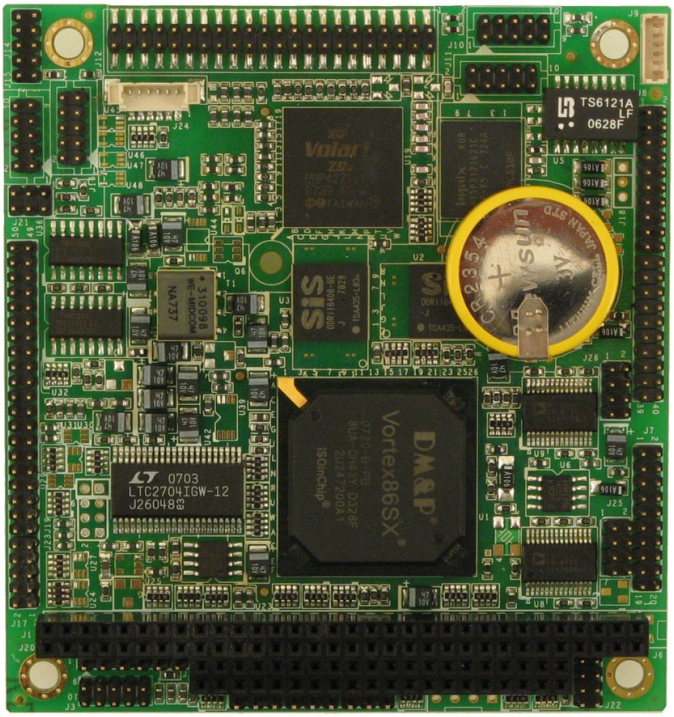
DIAMOND SYSTEMS CORPORATION

Helios User Manual

300-800MHz PC/104 Single Board Computer
with Integrated Data Acquisition

User Manual v1.01



© Copyright 2009
Diamond Systems Corporation
1255 Terra Bella Ave.
Mountain View, CA 94043
Tel (650) 810-2500
Fax (650) 810-2525
www.diamondsystems.com

Table of Contents

Introduction....7

Feature Overview....7
Processor, Memory, Buses....7
Analog I/O....7
Digital I/O....7
Data Acquisition....8
Counter/Timers....8
Video Features....8
Ethernet....8
Standard Peripheral Interfaces....8
Bus Interfaces....8
Power Supply....9
Battery Backup....9
Watchdog Timer....9
Functional Block Diagram....10
Board Diagram....11
Connector Summary....12
Jumper Summary....12

Connectors....13

PC/104 ISA Bus (J1, J2)....13
PS/2 Mouse and Keyboard (J3)....14
Input Power (J4)....14
I/O Power (J5)....15
External Battery (J6)....15
Digital I/O (J7)....16
Serial Port I/O (J8)....16
RS-232 Pin Assignments....17
RS-485 Pin Assignment....18
RS-422 Pin Assignment....19
LCD Backlight (J9)....19
VGA (J10)....20
Ethernet (J11)....21
IDE (J12)....21
LCD Panel - LVDS Interface (Bottom of J12)....23
Auxiliary (J14)....24
USB 2.0 (J15, J16)....25
Data Acquisition I/O (J17)....25
Auto-calibration (Data acquisition only) (J19)....27
Standard JTAG Configuration Interface (Data acquisition only) (J20)....27
Panel Power Input (J22)....28

Board Configuration Jumpers....29

DAQ Interrupt Configuration (J21)....29
Interrupt Pull-down Resistor Sharing....29

IRQ Selection....30
DAC Configuration (J23)....30
Single-ended/Differential Input Settings....30
Unipolar/Bipolar Input Settings....31
RS-422/RS-485 Configuration (J25, J26)....31
RS-422/RS-485 Termination....32
Multiport Receiver Termination....32
Erase CMOS RAM (J27)....32
Board Configuration Resistors....33

Installation and Configuration....34

Hardware Installation....34
General Setup....34
IDE Configuration....34
DOS Operating System Installation....34
CompactFlash Issues under DOS....35

Boot Procedures....36

Booting into MS-DOS, FreeDOS or ROM-DOS....36
Booting into Linux....36

BIOS Setup....37

BIOS Settings....37
Serial Ports....37
Parallel Port....37
LCD Video Settings....37
Miscellaneous Settings....37
BIOS Console Redirection Settings....38

System I/O Description....39

Ethernet....39
Serial Ports....39
PS/2 Ports....39
USB Ports....39
System Resources....40
Console Redirection to a Serial Port....40
Watchdog Timer....41
Flash Memory....41
Backup Battery....41
System Reset....42
On-Board Video....42

Data Acquisition Circuit....43

Data Acquisition Circuitry I/O Map....43
Overview....43
Register Map Page Summary....44
Register Map Bit Summary....45

Page 0 Register Definitions....46
Page 1 Register Definitions....58
Page 2 Register Definitions....62

Analog-to-Digital Input Ranges and Resolution....66

Overview....66
Input Range Selection....66
Input Range Table....66

Performing an A/D Conversion....67

Introduction....67
Select the Input Channel....67
Select the Input Range....67
Wait for Analog Input Circuit to Settle....67
Perform an A/D Conversion on the Current Channel....67
Wait for the Conversion to Finish....68
Read the Data from the Board....68
Convert the numerical data to a meaningful value....68
Conversion Formula for Bipolar Input Ranges....69
Conversion Formula for Unipolar Input Ranges....69

A/D Scan, Interrupt and FIFO Operation....71

Digital-to-Analog Output Ranges and Resolution....73

Description....73
Resolution....73
Output Range Selection....73
D/A Conversion Formulas and Tables....73
D/A Conversion Formulas for Unipolar Output Ranges....74
D/A Conversion Formulas for Bipolar Output Ranges....74

Generating an Analog Output....76

Compute the D/A Code for the Desired Output Voltage....76
Write the Value to the Selected Output Channel Registers....76
Wait for the D/A to Update....76

Analog Circuit Calibration....78

Digital I/O Operation....79

Programming the Watchdog Timers Using the DSCUD API....79
dscDIOSetConfig Function....80
dscDIOOutputByte Function....81
dscDIOInputByte Function....81
dscDIOInputBit Function....82
dscDIOOutputBit Function....82

Counter/Timer Operation....83

Counter 0 – A/D Sample Control....83
Counter 1 – Counting/Totalizing Functions....83
Command Sequences....84
Load and Enable (Run) a Counter Sequence....84
Read a Counter Sequence....84
Disabling the Counter Gate Command....85
Clearing a Counter Sequence....85

Watchdog Timer Programming....86

Programming the Watchdog Timers Using the DSCUD API....86
dscWatchdogEnable Function....87
dscWatchdogDisable Function....89
dscWatchdogTrigger Function....89
Watchdog Timer Programming Example....89

FlashDisk Module....91

Installing the FlashDisk Module....91
Configuration....91
Using the FlashDisk with Another IDE Drive....92
Power Supply....92

FlashDisk Programmer Board....93

Panel I/O Board....94

I/O Cables....95

Specifications....96

CPU....96
Analog Inputs....96
Analog Outputs....96
Digital I/O....96
Counter/Timers....97
Power Supply....97
General....97
Additional Information....97
Technical Support....97

Figures

Figure 1: Helios Functional Block Diagram....10

Illustration 2: 10

Figure 3: Helios Board Layout....11

Figure 4: J21 Jumper Block....29

Figure 5: Interrupt Pull-down Resistor Sharing Jumper....29

Figure 6: DAQ IRQ Selection Jumpers.....30
Figure 7: J23 Jumper Block.....30
Figure 8: Single-ended/Differential Input Jumper.....31
Figure 9: Unipolar/Bipolar Input Settings Jumpers.....31
Figure 10: J25/J26 Jumper Blocks.....31
Figure 11: RS-485 Selection Jumper.....32
Figure 12: RS-422 Selection Jumper.....32
Figure 13: Helios Data Acquisition Block Diagram.....43
Figure 14: FlashDisk Module.....91
Figure 15: FlashDisk Programmer Board Layout.....93
Figure 16: Helios Panel I/O Board.....94
Figure 17: Helios Cables Kit (C-HLV-KIT).....95

Introduction

Helios is an embedded single board computer in the PC/104 small form factor that integrates a complete embedded PC and data acquisition circuitry into a single board.

The single board Helios computer is a Pentium III class device with on-board central processing, memory and memory management devices and I/O management for specific functions. The Helios board,

- Communicates externally over the ISA bus and I/O ports.
- Generates on-board RGB video for CRT display systems.
- Contains LVDS formatting to drive a flat panel
- Is powered from an externally regulated +5VDC supply.

The Helios CPU uses the ISA bus, internally, to connect serial ports 1 through 4 and the data acquisition circuit to the processor. The ISA bus is brought out to an expansion connector to mate with add-on boards. Diamond Systems manufactures a wide variety of compatible PC/104 add-on boards for analog I/O, digital I/O, counter/timer functions, serial ports and power supplies.

As shown in the following table, various video, digital I/O and data acquisition options are available.

Model	***VGA/LCD Video***	***Digital I/O***	***Data Acquisition***
HLV800-256DV	Yes	16 lines	No
HLV800-256AV	Yes	40 lines	Yes
HLV300-128DV	Yes	16 lines	No

Feature Overview

The Helios board includes the following key system and data acquisition features.

Processor, Memory, Buses

- 300-800MHz Vortex86DX/SX CPU with integrated North Bridge/South Bridge, with embedded BIOS.
- 256MB DDR2 RAM system memory.
- PC/104 (ISA) bus interface.
- Flash controller.
- 33MHz PCI Bus.

Analog I/O

- 16 single-ended or 8 differential analog inputs, 16-bit resolution.
- 100KHz maximum aggregate A/D sampling rate.
- Programmable input ranges/gains.
- Both bipolar and unipolar input ranges.
- Internal and external A/D triggering.
- 512-byte sample FIFO with programmable threshold.
- Auto-calibration.
- Four analog outputs, 12-bit resolution.
- ±5V, ±10V and 0-5V, 0-10V output ranges available.

Digital I/O

- Up to 40 programmable digital I/O lines.
- Enhanced output current capability: -8/+12mA maximum.

Data Acquisition

On board I^2C flash EEROM is provided for auto-calibration value storage.

Counter/Timers

- One 24-bit counter/timer for A/D sampling rate control.
- One 16-bit counter/timer for user counting and timing functions.
- Programmable gate and count enable.
- Internal (10MHz) or external clock source.

Video Features

Video circuitry is provided by the Volari chipset.

Ethernet

The board provides 10/100Base-T Ethernet integrated in the CPU. Magnetics are included on the board so that a complete circuit is provided.

Standard Peripheral Interfaces

- Four serial ports, 115.2kbaud max.
    - Two 16550-compatible RS-232 ports.
    - Two 16850-compatible ports with 128-byte FIFOs. These ports provide RS-232, RS-422 and automatic RS-485 half-duplex capability with RS-422/RS-485 termination.
- Four USB 2.0 ports.
- IDE UDMA-100 port, solid-state Flashdisk interface.
- PS/2 keyboard and mouse ports.

Helios contains four serial ports. Each port is capable of transmitting at speeds of up to 115.2Kbaud, and uses a dedicated RS-232 transceiver with ESD protection.

Ports COM1 and COM4 are built into the standard chipset, consisting of standard 16550-type UARTs with 16-byte FIFOs. COM1 and COM2 can also be BIOS-selected for RS-232, RS-422 or RS-485. Termination resistors of 120 ohms can be jumper-enabled on these two ports.

Console redirection feature is incorporated. This feature enables keyboard input and character video output to be routed to one of the serial ports.

The board contains provision for mounting a solid state IDE flash disk module with capacities ranging from 32MB and greater. The module mounts onto the board using a 44-pin 2mm pitch header and a hold-down mounting hole with spacer and screws.

Bus Interfaces

The PCI bus is generated by the Vortex86DX/SX processor module and is used internally for the Ethernet circuit. The PCI bus is not brought out to a PCI-104 expansion connector.

The South Bridge also provides the ISA bus, which is extended to the PC/104 interface and provides the following types of I/O.

- Dual UART for 2 serial ports.
- Data acquisition circuit, including a watchdog timer, analog and digital I/O, and two counter/timers

Power Supply

The power supply is an on-board DC-DC converter, allowing an input range of +5VDC, ±5%. Jumper selection allows power to be taken from the PC-104 bus and not from the on-board converter.

The power supply includes ATX power switching and ACPI power management support. The master +5V input is controlled by the ATX function with an external switch input.

Battery Backup

Helios contains a backup battery for the real-time clock and BIOS settings. The battery is directly soldered to the board and provides a minimum 7 year backup lifetime at 25°C.

The on-board battery may be bypassed with a jumper or replaced with an external battery connected to an external battery connector.

Watchdog Timer

A watchdog timer (WDT) circuit consists of two software-programmable timers.

Functional Block Diagram

Figure 1 shows the Helios functional block diagram.

Figure 1: Helios Functional Block Diagram

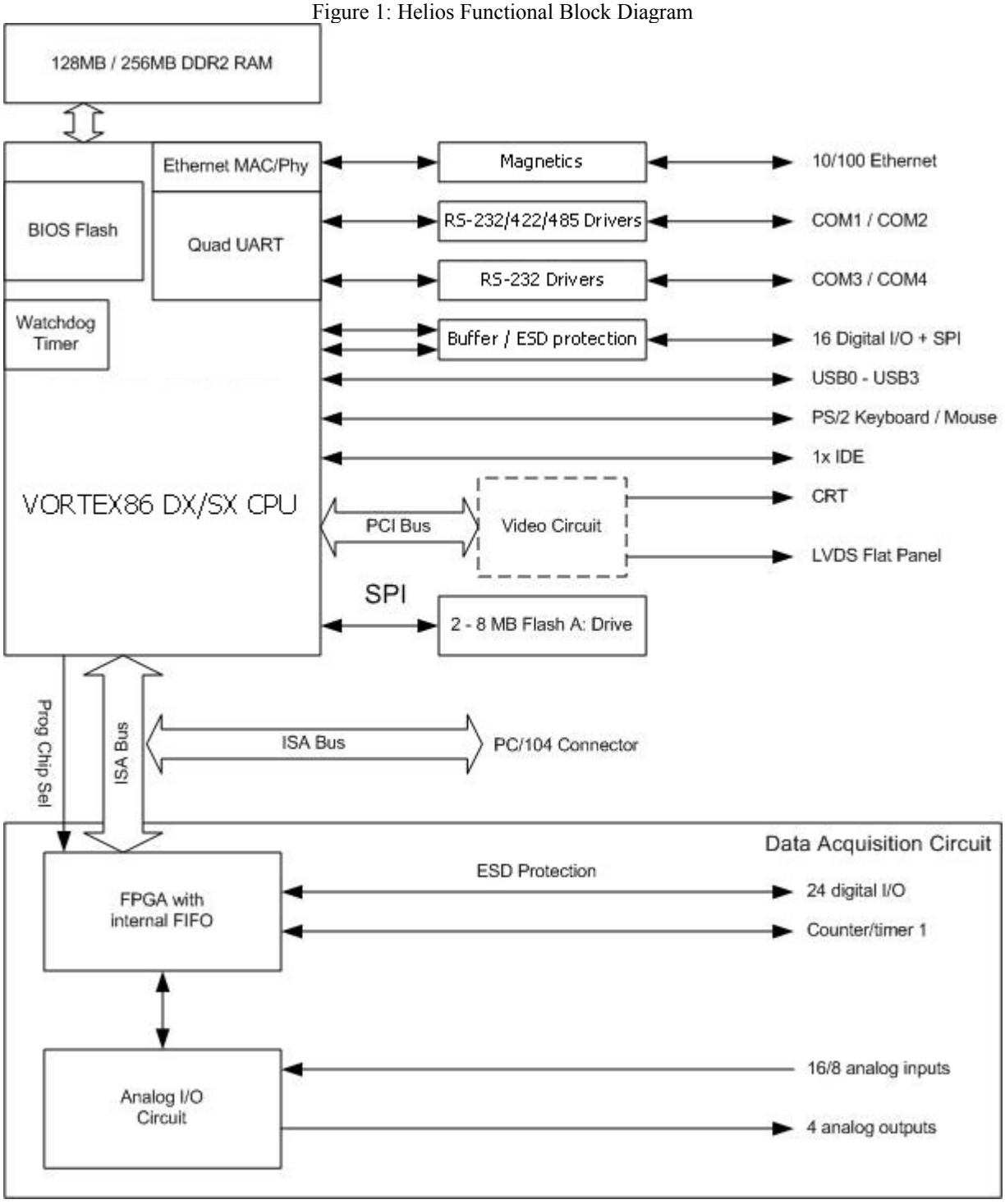
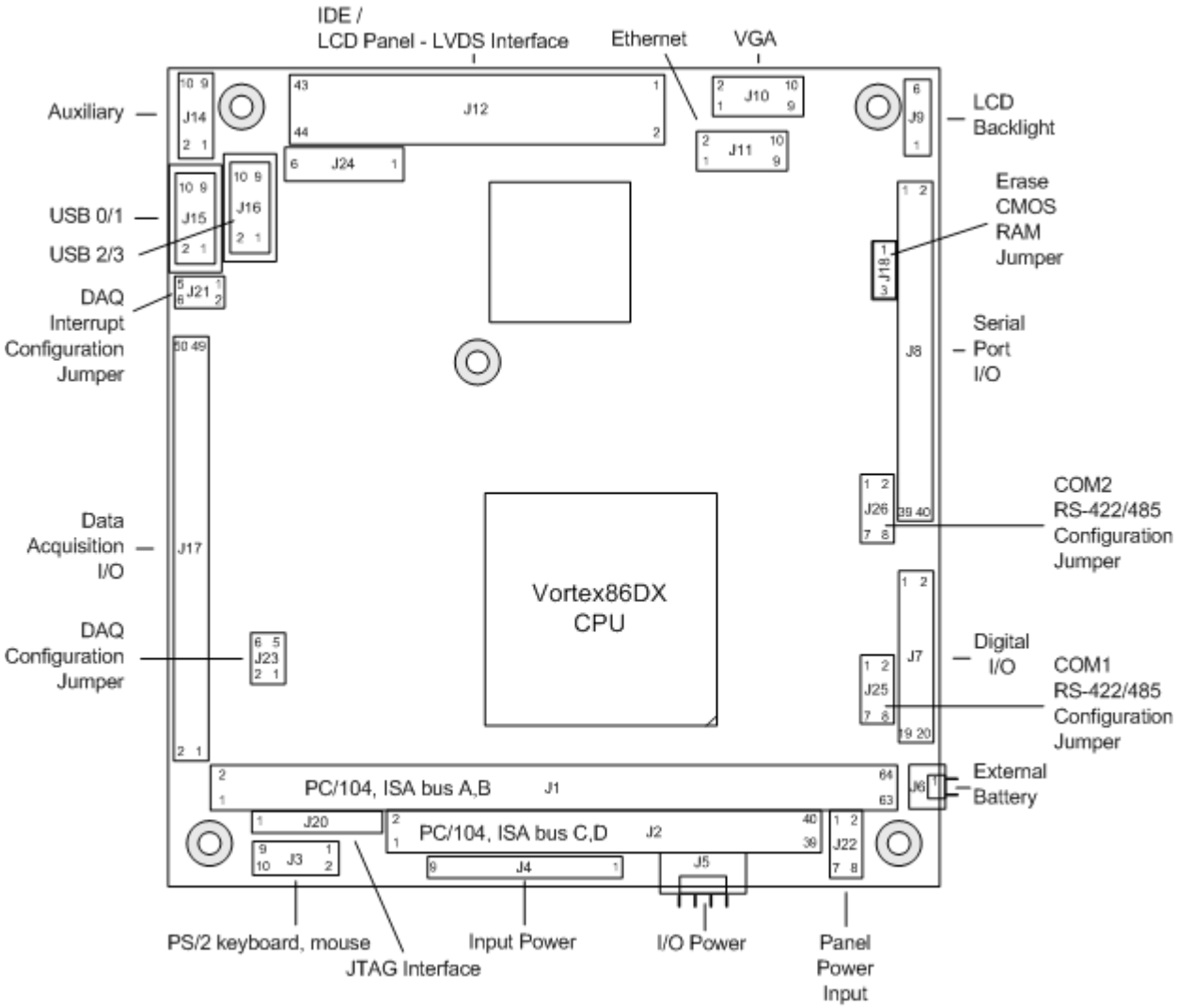


Illustration 2:

Board Diagram

Figure 3 shows the Helios SBC board layout, including connectors, jumper blocks and mounting holes.

Figure 3: Helios Board Layout



Connector Summary

The following table lists the connectors on the Helios board.

Connector	*Description*
J1	PC/104, ISA bus A,B
J2	PC/104, ISA bus C,D
J3	PS/2 keyboard/mouse
J4	Input power
J5	I/O power
J6	External battery
J7	Digital I/O
J8	Serial port I/O
J9	LCD backlight
J10	VGA
J11	Ethernet
J12	IDE and LCD Panel - LVDS Interface
J14	Auxiliary
J15	USB 0/1
J16	USB 2/3
J17	Data acquisition I/O
J19	Auto-calibration
J20	Standard JTAG Configuration Interface
J22	Panel power input

Jumper Summary

The following table lists the jumpers on the Helios board.

Jumper	*Description*
J18	LCD backlight
J21	DAQ interrupt configuration
J23	DAQ configuration
J25	COM1 RS-422/RS-485 configuration
J26	COM2 RS-422/RS-485 configuration
J27	Erase CMOS RAM

Connectors

This section describes the on-board Helios connectors.

Note: Pins marked as "key" are cut away or removed, unless otherwise indicated.

PC/104 ISA Bus (J1, J2)

Connectors J1 and J2 carry the ISA bus signals. The following diagram shows the PC/104 A and B pin layout for J1 and the C and D pin layout for J2.

J1

IOCHCHK-	A1	B1	Ground
SD7	A2	B2	RESETDRV
SD6	A3	B3	+5V
SD5	A4	B4	IRQ9
SD4	A5	B5	-5V
SD3	A6	B6	DRQ2
SD2	A7	B7	-12V
SD1	A8	B8	ENDXFR-
SD0	A9	B9	+12V
IOCHRDY	A10	B10	Key
AEN	A11	B11	SMEMW-
SA19	A12	B12	SMEMR-
SA18	A13	B13	IOW-
SA17	A14	B14	IOR-
SA16	A15	B15	DACK3-
SA15	A16	B16	DRQ3
SA14	A17	B17	DACK1-
SA13	A18	B18	DRQ1
SA12	A19	B19	REFRESH-
SA11	A20	B20	SYSCLK
SA10	A21	B21	IRQ7
SA9	A22	B22	IRQ6
SA8	A23	B23	IRQ5
SA7	A24	B24	IRQ4
SA6	A25	B25	IRQ3
SA5	A26	B26	DACK2-
SA4	A27	B27	TC
SA3	A28	B28	BALE
SA2	A29	B29	+5V
SA1	A30	B30	OSC
SA0	A31	B31	Ground
Ground	A32	B32	Ground

J2

Ground	C0	D0	Ground
SBHE-	C1	D1	MEMCS16--
LA23	C2	D2	IOCS16-
LA22	C3	D3	IRQ10
LA21	C4	D4	IRQ11
LA20	C5	D5	IRQ12
LA19	C6	D6	IRQ15
LA18	C7	D7	IRQ14
LA17	C8	D8	DACK0-
MEMR-	C9	D9	DRQ0
MEMW-	C10	D10	DACK5-
SD8	C11	D11	DRQ5
SD9	C12	D12	DACK6-
SD10	C13	D13	DRQ6
SD11	C14	D14	DACK7-
SD12	C15	D15	DRQ7
SD13	C16	D16	+5
SD14	C17	D17	MASTER-
SD15	C18	D18	Ground
Key	C19	D19	Ground

PS/2 Mouse and Keyboard (J3)

Connector J3 provides the standard PS/2 keyboard and mouse signals.

+5Vin	1	2	Key
KB Data	3	4	MS Data
KB Clock	5	6	MS Clock
Ground	7	8	Key
Ground	9	10	+5Vin

Signal	***Definition***
+5Vin	keyboard PS/2 pin 4
KB Data	keyboard PS/2 pin 1
KB Clock	keyboard PS/2 pin 5
MS Data	mouse PS/2 pin 1
MS Clock	mouse PS/2 pin 5
Ground	PS/2 pin 3

Input Power (J4)

Input power may be supplied using either J4, the I/O power connector (J5), an external supply, or directly through the PC/104 bus power pins, if a PC/104 power supply is used with the CPU.

The board only requires +5VDC input power to operate. All other required voltages are generated on board. However, the PC/104 bus (J1) may be used to supply ±5V and ±12V, if needed.

Multiple +5V and ground pins are provided for extra current carrying capacity, if needed. Each pin is rated at 3A max. For applications requiring less than 3A, the first four pins may be connected to a standard 4-pin miniature PC power connector, or the alternate power I/O connector may be used. For a larger PC/104 stack the total power requirements should be calculated to determine whether additional wires are necessary.

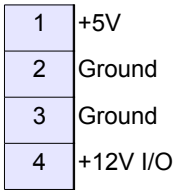
1	+5V In
2	Ground
3	Key
4	+12V In
5	Ground
6	+5V In
7	-12V In
8	-5V In
9	+3.3V

Signal	*Definition*
+3.3V	+3.3V input power for LCD.
+5V	+5V input. Only +5VDC power is required for board operation.
+12V	+12V input.
Ground	Ground

I/O Power (J5)

Connector J5 provides an alternate connector for either input power to the system or output power for use with external drives.

This connector mates with Diamond Systems cable part number 6981006, which provides a standard full-size power connector for a hard drive or CD-ROM drive and a standard miniature power connector for a floppy drive.



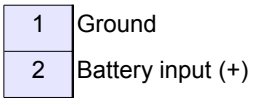
Signal	*Definition*
+5V	This is provided by the on-board power supply, derived from the input power. It is switched off when the board is powered down.
+12V I/O	This is provided by the 12V input pin on the main power connector. It is switched off when the board is powered down.
Ground	These are 0V ground references for the power output voltage rails, above.

External Battery (J6)

Connector J6 is used to connect an external battery to augment or replace the on-board backup battery.

In addition to the external battery connected to J6, the on-board battery is another possible sources for maintaining the Real-Time Clock and the CMOS BIOS settings for various system configurations. The battery that has the highest voltage will see the majority of the current draw, which is minimal.

Note: There must be a battery voltage input for the default power-up mode.



Signal	*Definition*
Battery Input	Battery input voltage. The battery voltage for this input should be 3V. The current draw averages under 5µA at 3V.
Ground	Battery ground.

Digital I/O (J7)

Connector J7 provides 16 digital I/O lines from the Vortex CPU. These lines are buffered and have ESD protection to protect the CPU from potential damage. The buffers enable the direction to be programmed in 8-bit groups, using two additional GPIO lines from the Vortex processor. The direction may be set in the BIOS or by programming the CPU I/O control registers.

DIOLCA0	1	2	DIOLCA1
DIOLCA2	3	4	DIOLCA3
DIOLCA4	5	6	DIOLCA5
DIOLCA6	7	8	DIOLCA7
DIOLCB0	9	10	DIOLCB1
DIOLCB2	11	12	DIOLCB3
DIOLCB4	13	14	DIOLCB5
DIOLCB6	15	16	DIOLCB7
+5V	17	18	Ground
Key	19	20	Ground

Signal	*Definition*
DIOLCA0-7	Vortex CPU GPIO port 0.
DIOLCB0-7	Vortex CPU GPIO port 1.
+5V	Power; connected to switched +5V.
Ground	Digital ground.

Serial Port I/O (J8)

Connector J8 provides access to the four serial ports of the Vortex CPU. The PORT1 and PORT2 ports are, independently, jumper-configurable for either RS-232, RS-485 or RS-422 protocol. Jumpers J25 and J26 are used to select the protocol. The PORT3 and PORT4 ports are fixed RS-232 protocol. All four serial ports can be independently enabled.

Connector pins are dedicated to a port, as shown in the following table.

Port No.	*Pin Assignment*
PORT1	Pins 1 - 10
PORT2	Pins 11 - 20
PORT3	Pins 21 - 30
PORT4	Pins 31 - 40

The following tables list the signals and associated DE-9 pin numbers for each of the protocols; pin assignment differs, depending on the protocol selected.

RS-232 Pin Assignments

COM1:	DCD1	1	2	DSR1
	RXD1	3	4	RTS1
	TXD1	5	6	CTS1
	DTR1	7	8	RI1
	Ground	9	10	NC
COM2:	DCD2	11	12	DSR2
	RXD2	13	14	RTS2
	TXD2	15	16	CTS2
	DTR2	17	18	RI2
	Ground	19	20	NC
COM3:	DCD3	21	22	DSR3
	RXD3	23	24	RTS3
	TXD3	25	26	CTS3
	DTR3	27	28	RI3
	Ground	29	30	N/C
COM4:	DCD4	31	32	DSR4
	RXD4	33	34	RTS4
	TXD4	35	36	CTS4
	DTR4	37	38	RI4
	Ground	39	40	NC

Signal	***Definition***	***DE-9 Pin***	***Direction***
DCD*n*	Data Carrier Detect	pin 1	Input
DSR*n*	Data Set Ready	pin 6	Input
RXD*n*	Receive Data	pin 2	Input
RTS*n*	Request to Send	pin 7	Output
TXD*n*	Transmit Data	pin 3	Output
CTS*n*	Clear to Send	pin 8	Input
DTR*n*	Data Terminal Ready	pin 4	Output
RI*n*	Ring Indicator	pin 9	Input
Ground	Ground	-	-

RS-485 Pin Assignment

Only J8 connector pins 21 through 40, PORT3 and PORT4, are used for RS-485.

COM1:	NC	21	22	NC
	TXD/RXD+3	23	24	TXD/RXD-3
	Ground	25	26	NC
	NC	27	28	NC
	Ground	29	30	NC
COM2:	NC	31	32	NC
	TXD/RXD+4	33	34	TXD/RXD-4
	Ground	35	36	NC
	NC	37	38	NC
	Ground	39	40	NC

Signal	***Definition***	***DE-9 Pin***	***Direction***
TXD/RXD+*n*	Differential Transceiver Data (HIGH)	pin 2	bi-directional
TXD/RXD-*n*	Differential Transceiver Data (LOW)	pin 7	bi-directional
Ground	Ground	-	-
NC	(not connected)	-	-

RS-422 Pin Assignment

Only J8 connector pins 21 through 40, PORT3 and PORT4, are used for RS-422.

COM1:	NC	21	22	NC
	TXD+1	23	24	TXD-1
	Ground	25	26	RXD-1
	RXD+1	27	28	NC
	Ground	29	30	NC
COM2:	NC	31	32	NC
	TXD+2	33	34	TXD-2
	Ground	35	36	RXD-2
	RXD+2	37	38	NC
	Ground	39	40	NC

Signal	***Definition***	***DE-9 Pin***	***Direction***
TXD+*n*/TXD-*n*	Differential transmit data	-	Output
RXD+*n*/RXD-*n*	Differential receive data	-	Input
Ground	Ground	-	-
NC	(not connected)	-	-

LCD Backlight (J9)

Connector J9 provides the backlight power and control for the optional LCD panel. See the description for connector J12 for detailed information on the LCD data interface.

The control signal is used to allow the system to power-down the backlight when the system enables monitor-power-down during power management control.

Note: If needed, 12V must be provide, either on one of the input power connectors or on the 12V pin (J1, B9) of the PC/104 connector, for the LCD backlight to operate. The board does not generate the voltage, internally.

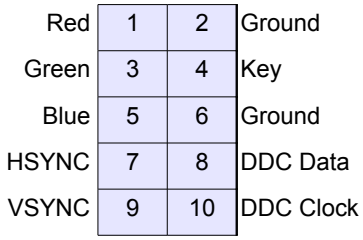
1	+12v
2	+12v
3	Ground
4	Ground
5	Enable
6	Brightness

Signal	*Definition*
+12V	+5V or +12V power supply for LCD Backlight assembly, jumper J18 selectable. The +12V supply is removed when the system is powered down.
Enable	Enable (GPIO output) 0 = disable. open circuit = enable. GP35 0 = disable. 1 = enable. (**Default**)
Brightness	Brightness, 0-5VDC. 0V = max. 5V = min. GP36 0 = max. (**Default**) 1 = min.
Ground	Ground for LCD Backlight assembly.

VGA (J10)

Connector J10 is used to connect a VGA monitor.

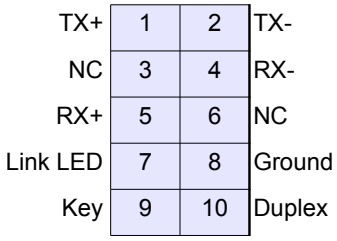
Note: While the DDC serial detection pins are present, a 5V power supply is not provided, and the legacy "Monitor ID" pins are also not used.



Signal	*Definition*
Red	RED signal (positive, 0.7Vpp into 75 Ohm load)
Green	GREEN signal (positive, 0.7Vpp into 75 Ohm load)
Blue	BLUE signal (positive, 0.7Vpp into 75 Ohm load)
DDC Clock/Data	Digital serial I/O signals used for monitor detection (DDC1 specification)
HSYNC	Horizontal sync
VSYNC	Vertical sync
Ground	Ground return

Ethernet (J11)

The 10/100 Base-T, full-duplex Ethernet interface is provided by connector J11.



Signal	*Definition*
TX+/TX-	Transmit data.
RX+/RX-	Receive data.
Link LED	Link activity indication; referenced to ground.
Duplex	
Ground	Ground.

IDE (J12)

The IDE connector, J12, is used to connect two IDE drives, including hard disks, CD-ROMs and Flashdisk modules.

This connector mates with Diamond Systems cable part number 6981004.

Reset -	1	2	Ground
D7	3	4	D8
D6	5	6	D9
D5	7	8	D10
D4	9	10	D11
D3	11	12	D12
D2	13	14	D13
D1	15	16	D14
D0	17	18	D15
Ground	19	20	Key
DRQ	21	22	Ground
IDEIOW-	23	24	Ground
IDEIOR-	25	26	Ground
IORDY	27	28	Ground
DACK-	29	30	Ground
IRQ14	31	32	Pulled low for 16-bit operation
A1	33	34	NC
A0	35	36	A2
CS0-	37	38	CS1-
LED-	39	40	Ground
+5v	41	42	+5v
Ground	43	44	NC

Signal	***Definition***
Reset -	Reset
D0-D15	16-bit data
Ground	Ground
DRQ	DDRQ
DACK-	DDACK
IDEIOW-	I/O write
IDEIOR-	I/O read
IORDY	IOC HRDY
IRQ14	IRQ
A0-A2	Address 0-2
CS0-	Chip select 1P
CS1-	Chip select 3P
LED-	Activity indication
+5V	+5VDC

LCD Panel - LVDS Interface (Bottom of J12)

Connector J12 provides access to the internal LVDS LCD display drivers.

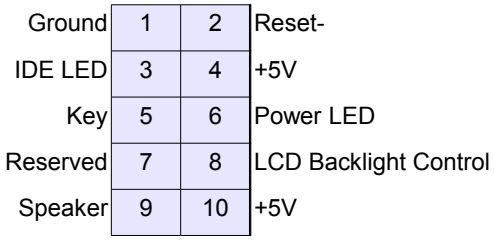
Note: The LCD also requires the backlight (J9) to be connected to function correctly.

1	Ground/D3+
2	Ground/D3-
3	SD
4	FRC
5	SigGround
6	PClk+
7	PClk-
8	SigGround
9	D2+
10	D2-
11	SigGround
12	D1+
13	D1-
14	SigGround
15	D0+
16	D0-
17	PwrGround
18	PwrGround
19	3.3V
20	3.3V

Signal	***Definition***
Ground/D3+	Ground or D3+, depending on video chip.
Ground/D3-	Ground or D3-, depending on video chip.
SD	Scan Direction. High = Reverse scan. Low/open = Normal scan.
FRC	Frame Rate Control. High = On. Low/open = Off.
PClk+	Pixel clock +.
PClk-	Pixel clock -.
D0+,D0-/D2+,D2-	
3.3V	VCC 3.3v or 5V (Jumper J18 configured).
SigGround	Signal ground.
PwrGround	Power ground.

Auxiliary (J14)

Connector J14 provides access to common auxiliary signals.



Signal	***Definition***
IDE LED	IDE drive activity indication LED.
Power LED	Power enabled LED.
LCD Backlight Control	User-provided brightness control for the LCD backlight. See the description for connector J9. 0V = max. 5V = min.
Speaker	Speaker connection; referenced to +5V.
+5V	+5VDC power.
Reset-	Connect this pin to ground to cause a reset condition.
Ground	Ground
Reserved	Reserved for future use.

USB 2.0 (J15, J16)

The board features four USB 2.0 ports. Connector J15 interfaces to USB port 0/1 and connector J16 interfaces to USB ports 2/3. USB 2.0 provides a 480Mbps maximum data transfer rate.

J15

Key	1	2	Shield
USB1 Pwr-	3	4	USB0 Pwr-
USB1 Data+	5	6	USB0 Data+
USB1 Data-	7	8	USB0 Data-
USB1 Pwr+	9	10	USB0 Pwr+

J16

Key	1	2	Shield
USB3 Pwr-	3	4	USB2 Pwr-
USB3 Data+	5	6	USB2 Data+
USB3 Data-	7	8	USB2 Data-
USB3 Pwr+	9	10	USB2 Pwr+

Signal	***Definition***
USB0-3 Pwr+	+5V power for USB ports 0-3
USB0-3 Pwr-	-5V power for USB ports 0-3
USB0-3 Data+	Data + for USB ports 0-3
USB0-3 Data-	Data - for USB ports 0-3
Shield	Ground; tied to system ground.

Data Acquisition I/O (J17)

Connector J17 is provided on board with the data acquisition option.

DAQ interrupt options are set using jumper J21. DAQ single-ended/unipolar/bipolar options are configured using jumper J23.

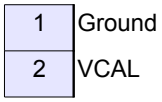
Diamond Systems cable number 6981163 provides a standard 50-pin connector at each end that mates with this connector.

DIO A0	1	2	DIO A1
DIO A2	3	4	DIO A3
DIO A4	5	6	DIO A5
DIO A6	7	8	DIO A7
DIO B0	9	10	DIO B1
DIO B2	11	12	DIO B3
DIO B4	13	14	DIO B5
DIO B6	15	16	DIO B7
DIO C0	17	18	DIO C1
DIO C2	19	20	DIO C3
DIO C4/GATE0	21	22	DIO C5/GATE1
DIO C6/CLK1	23	24	DIO C7/OUT0
EXTTRIG	25	26	TOUT1
+5V out	27	28	DGND
VOUT0	29	30	VOUT1
VOUT2	31	32	VOUT3
AGND(Vout)	33	34	AGND(Vin)
VIN0	35	36	VIN8
VIN1	37	38	VIN9
VIN2	39	40	VIN10
VIN3	41	42	VIN11
VIN4	43	44	VIN12
VIN5	45	46	VIN13
VIN6	47	48	VIN14
VIN7	49	50	VIN15

Signal	***Definition***
DIO A7-A0	Digital I/O port A; programmable direction.
DIO B7-B0	Digital I/O port B; programmable direction.
DIO C7-C0	Digital I/O port C; programmable direction. C7-C4 may be configured for counter/timer signals.
EXTTRIG	External A/D trigger input.
TOUT1	Counter/Timer 1 output.
Vin 7/7+ ~ Vin 0/0+	Analog input channels 7 – 0 in single-ended mode. High side of input channels 7 – 0 in differential mode.
Vin 15/7- ~ Vin 8/0-	Analog input channels 15 – 8 in both single-ended mode. Low side of input channels 7 – 0 in differential mode.
VOUT0-3	Analog output channels 0 – 3.
+5V out	Connected to switched +5V supply (Output only! Do not connect to external supply).
DGND	Digital ground (0V - reference); used for digital circuitry only.
AGND	Analog ground; used for analog circuitry only. Vout pin is for analog outputs, Vin pin is for analog inputs.

Auto-calibration (Data acquisition only) (J19)

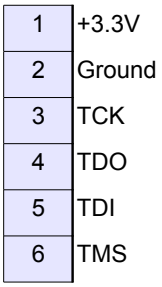
Connector J19 is used for measurement of on-board reference voltages. A precision meter is connected to this connector during reference measurement and should be disconnected before performing auto-calibration.



Signal	*Definition*
VCAL	Reference voltage.
Ground	Ground

Standard JTAG Configuration Interface (Data acquisition only) (J20)

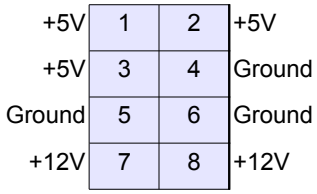
Connector J20 is the JTAG configuration interface for factory use and firmware upgrade.



Signal	*Definition*
+3.3V	Power
Ground	Ground
TCK	Test clock input
TDO	Test data output
TDI	Test data input
TMS	Test mode select

Panel Power Input (J22)

Connector J22 provides power to the board when connected to the I/O panel board. All signals are routed to their corresponding pins on the PC/104 connector.



Signal	***Definition***
+5V	+5V input. Only +5VDC power is required for board operation.
+12V	+12V input. Provided as a pass-through to the PC/104 bus connector (J1).
Ground	Ground

Board Configuration Jumpers

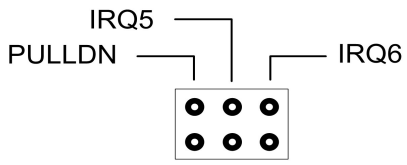
The following jumpers are defined for configuring the board.

Jumper Block	*Configuration Functions*
J18	LCD backlight.
J21	DAQ interrupt configuration.
J23	DAQ configuration.
J25	COM1 RS-422/RS-485 configuration.
J26	COM2 RS-422/RS-485 configuration.
J27	Erase CMOS RAM.

DAQ Interrupt Configuration (J21)

Jumper block J21 is used to configure DAQ interrupt pull-down resistor sharing and IRQ selection.

Figure 4: J21 Jumper Block

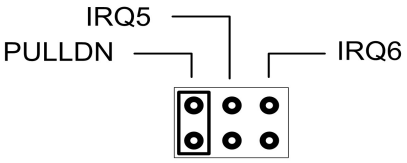


Jumper Label	*Function*
PULLDN	Pulls down IRQ.
IRQ5	Selects IRQ5.
IRQ6	Selects IRQ6.

Interrupt Pull-down Resistor Sharing

Configure the jumper as shown in the following figure to share the IRQ selection.

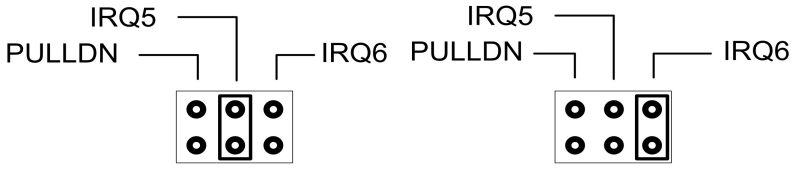
Figure 5: Interrupt Pull-down Resistor Sharing Jumper



IRQ Selection

Configure the jumper as shown in the following figure to select IRQ5 or IRQ6. The IRQ selection is mutually exclusive.

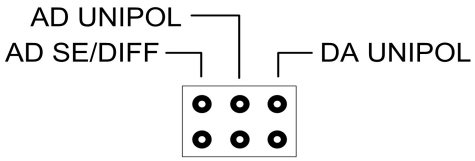
Figure 6: DAQ IRQ Selection Jumpers



DAC Configuration (J23)

Jumper block J23 is used to configure the A/D and D/A circuits.

Figure 7: J23 Jumper Block



Jumper Label	***Function***
AD SE/DIFF	A/D single-ended/differential selection.
AD UNIPOL	A/D unipolar/bipolar selection.
DA UNIPOL	D/A unipolar/bipolar selection.

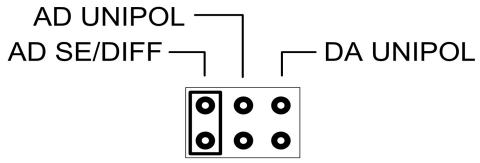
Single-ended/Differential Input Settings

Helios can accept both single-ended and differential inputs. A single-ended input uses two wires: input and ground. The measured input voltage is the difference between these two wires. A differential input uses three wires: input(+), input(–) and ground. The measured input voltage is the difference between the (+) and (–) inputs.

Differential inputs are frequently used either when the grounds of the input device and the measurement device (Helios) are at different voltages, or when a low-level signal is being measured that has its own ground wire. A differential input also has higher noise immunity than a single-ended input because most noise affects both (+) and (–) input wires equally, so the noise is canceled out in the measurement. The disadvantage of differential inputs is that only half as many are available because two input pins are required to produce a single differential input.

Helios can be configured for either 16 single-ended inputs, or eight differential inputs, as shown below. The default setting is single-ended mode.

Figure 8: Single-ended/Differential Input Jumper



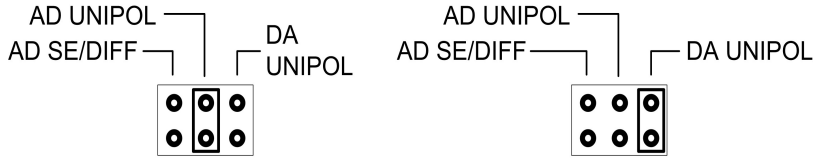
If you have a combination of single-ended and differential input signals, select differential mode. Then, to measure the single-ended signals, connect the signal to the plus (+) input and connect analog ground to the minus (–) input.

> **WARNING: The maximum range of voltages that can be applied to an analog input on Helios without damage is ±35V. If you connect the analog inputs on Helios to a circuit whose ground potential plus maximum signal voltage exceeds ±35V, the analog input circuit may be damaged. Check the ground difference between the input source and Helios before connecting analog input signals.**

Unipolar/Bipolar Input Settings

The analog inputs can be configured for either unipolar (positive input voltages only), or bipolar (both positive and negative input voltages). For unipolar inputs, install a jumper as shown in the following figure. For bipolar inputs, omit the jumper. The default configuration is bipolar mode (jumper out).

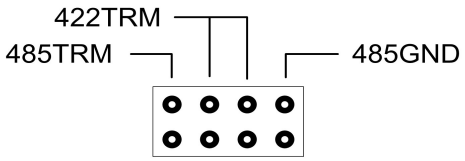
Figure 9: Unipolar/Bipolar Input Settings Jumpers



RS-422/RS-485 Configuration (J25, J26)

Use jumper J25 to select the COM1 RS-422/RS-485 termination and jumper J26 to select the COM2 RS-422/RS-485 termination.

Figure 10: J25/J26 Jumper Blocks

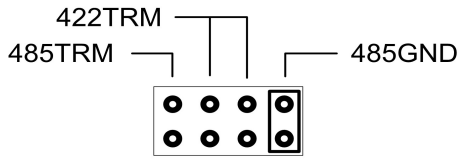


Jumper Label	*Function*
485TRM	RS-485/RS-422 Tx termination.
422TRM	RS-422 Rx termination.
485GND	RS-485 ground.

RS-422/RS-485 Termination

Jumper J25/J26 as shown in the following figure to terminate RS-422 or RS-485.

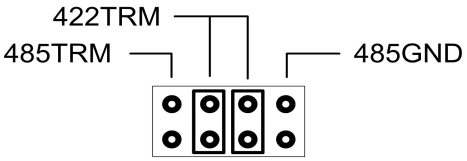
Figure 11: RS-485 Selection Jumper



Multiport Receiver Termination

Jumper J25/J26 as shown in the following figure for multi-port termination.

Figure 12: RS-422 Selection Jumper



Erase CMOS RAM (J27)

With the jumper in place the CPU powers up with the default BIOS settings.

Follow these steps to clear the CMOS RAM.

1. Power-down the CPU.
2. Remove the BAT jumper and move it to the battery disconnected position.
3. Wait a few seconds.
4. Insert the BAT jumper.
5. Power-up the CPU.

Note: Before erasing CMOS RAM, write down any custom BIOS settings.

Board Configuration Resistors

Resistors are used to configure the following functions.

- RS-422/RS-485 line termination enable.
- RS-422/RS-485 pull-up/pull-down enable.
- LCD voltage select.
- LCD backlight enable.
- LCD backlight brightness.
- VGA/LCD display type.

Installation and Configuration

This section describes the steps needed to get your Helios board up and running, and assumes that you have also purchased the Helios Development Kit. The development kit includes all cables described in the previous section, a power supply, USB floppy drive, mounting hardware, IDE flashdisk and the flashdisk programmer board.

Hardware Installation

General Setup

This section describes the initial setup procedures, which are identical regardless of which operating system or IDE configuration you are using.

1. Remove the Helios board from its packaging.

2. Install the mounting kit standoffs into the PC/104 mounting holes located at each corner of the board. This ensures that the board will not touch the surface beneath it, and helps redistribute the force when you push connectors onto the board.

3. Attach the VGA cable, 6981178, to connector J10. Connect your monitor VGA cable to the DB9 socket.

4. Take the power supply out of its packaging. (Do not plug it into the wall yet). Plug the 9-pin connector into the J4 connector on the board, immediately below the PC/104 bus. Be sure the red wire, +5 VDC, goes to pin 1.

5. (Optional for USB Devices) You will need to connect the USB cables if you are going to use a USB floppy, keyboard or mouse. Plug USB cable 6981082 into connector J15. If you need 3 or 4 USB sockets, connect cable 6981082 to connector J16.

IDE Configuration

Helios has a single IDE channel that can support up to two devices simultaneously (Master and Slave). IDE devices connect through J12, which is a 44-pin, laptop IDE connector. The following are a few example setups.

1. Connect one IDE flashdisk connected directly to J12.

2. Connect one laptop IDE hard drive directly to J12 through a 44-pin ribbon cable. This cable is available in the cable kit.

3. Use cable 6981004 to connect an IDE flashdisk programmer board to J12. You can then connect other 40-pin or 44-pin IDE compatible devices to the programmer board. Use cable 6981006, attached to J5, to provide power from the Helios board to 40-pin devices. Remember, the Helios cannot generate 12VDC. You will need to supply your own 12VDC line to the IDE device, or through the Helios power input connector.

DOS Operating System Installation

User the following sequence to install DOS operating systems: MS-DOS, FreeDOS and ROM-DOS.

1. Enable the following in BIOS:

   - Floppy Drive detection.
   - Legacy USB support.

2. Change the BIOS boot sequence so the system boots through the USB floppy drive.

3. Insert the DOS installation floppy disk into the USB floppy drive and start/restart the system.

4. Install any drivers needed.

Notes:

1. For DOS Ethernet, set *Operating System* to *other* in the BIOS.

2. DOS Sound emulation is currently not functional.

CompactFlash Issues under DOS

CompactFlash is incompatible with some utilities, under some versions of DOS.

- CompactFlash with ROM-DOS

  The ROM-DOS FDISK utility does not work with CompactFlash drives. The ROM-DOS FORMAT and SYS do work, however. If CompactFlash already has a DOS partition, the ROM-DOS utilities can be used to FORMAT the CompactFlash and install operating system files on CompactFlash.

- CompactFlash with FreeDOS

  The FreeDOS FDISK or FORMAT utility do not work with CompactFlash. However, the FreeDOS SYS utility is functional with CompactFlash.

- CompactFlash with MS-DOS

  The MS-DOS FDISK, FORMAT, and SYS utilities are not functional when used with CompactFlash. The MS-DOS operating system files cannot be installed on CompactFlash flash.

Boot Procedures

Booting into MS-DOS, FreeDOS or ROM-DOS

This section describes how to boot into a DOS-based operating system using a bootable floppy disk.

1. Plug the USB floppy drive into one of the USB terminals of cable 6981082.
2. Insert your DOS-based boot disk into the USB floppy drive.
3. Connect the power supply to the wall (to provide power to Helios).
4. At this point the Helios will boot and you should see the BIOS power-on self test. Press F2 to enter BIOS configuration.
5. Under the "Advanced" menu, scroll to "Legacy USB Support" and enable it. (Without enabling this option, the BIOS will not boot from a disk in the USB floppy drive).
6. Reboot the system to boot from your floppy disk.

Booting into Linux

This section describes how to setup the Helios board in preparation for a Linux install, from an installation CD-ROM onto a laptop IDE hard drive.

1. Connect the IDE FashDisk programmer board to J12.
2. Connect a CD-ROM drive jumpered for the slave position to the IDE FlashDisk programmer board through the 40-pin cable.
3. Connect the CD-ROM drive using cable 6981006 attached to J5. Be sure that an external 12VDC source is being suppled to J4, the CD-ROM.
4. Connect a laptop hardvdrive jumpered for master position to the second slot of the 44-pin cable.
5. Boot the Helios by plugging the power supply into the wall.
6. Press F2 at the power-on self test to go to the BIOS configuration screen.
7. Go to the "Boot" menu and confirm that the CD-ROM drive is first boot device.
8. Insert the boot CD for your operating system into the CD-ROM drive.
9. Save the BIOS settings and reboot.
10. You should now be able to install your OS.

BIOS Setup

Helios uses a BIOS from AMI modified to support the custom features of the Helios board.

BIOS Settings

To change the following BIOS settings, press F2 during system startup power on self-test (POST).

Serial Ports

The address and interrupt settings for serial ports COM1 and COM2 may be modified. COM1 and COM2 address and interrupt settings are configured using the Advanced, Advanced Chipset Control, I/O Chip Device Configuration menu.

Parallel Port

The parallel port is configured using the Advanced, I/O Chip Device Configuration menu. The port is set by default to ECP mode and located at address 0x378, IRQ 7 and DMA 3.

LCD Video Settings

Helios provides direct digital support for LVDS-based LCD interfaces only. As such, there are two settings that affect this support during BIOS boot.

- Boot Video Device – By default, this is set to "AUTO". With the AUTO setting, the system attempts to identify an RGB monitor (via DDC). If no RGB monitor is detected, the system enables LCD support. If you choose to use the LCD display regardless of standard monitor connection (i.e., with both connected at once), set "Boot Video Device" to "Both".
- Panel Type – This setting defaults to "7". Do not alter this setting unless specifically instructed to do so. This setting affects the LCD display modes supported; mode "7" is the only setting currently supported. Not all LCD displays are supported.

Miscellaneous Settings

- Memory Cache

  Unless there is a specific reason to change these settings, it is best to keep these settings as-is. Certain system functions, such as USB keyboard support under BIOS menus, may be adversely affected by changes to these settings. These cache settings can make a noticeable difference for low-level BIOS calls and, as such, can severely limit performance if they are disabled.

- Advanced Chipset Control

  the following settings should be retained:

  Frame Buffer Size: 8MB

  AGP Rate: 4X

  Expansion Bus Performance: Normal

  The Frame Buffer size can be increased for specific applications. Be aware, however, that an increase in this memory size will result in a decrease in overall system memory available. The AGP rate affects internal video accesses and does not affect any external bus speeds.

  "Expansion Bus Performance" is an adjustment to allow an increase in ISA I/O Access speeds. For applications where ISA I/O accesses seem to be a limiting factor, this performance may be increased to "Accelerated". Be aware that increasing these timings may adversely affect system stability with external add-on PC/104 cards. This setting has no direct affect on PCI or memory speeds; it only affects ISA PC/104 devices. It is best to leave this setting at "Normal," if there are no ISA I/O performance issues.

- Advanced

  Installed O/S: Win98

  Large Disk Access Mode: DOS

- On-Chip Multifunction Device

  USB Device: Enabled

  Legacy Audio: Disabled

  "Legacy Audio" only affects DOS-based applications when used with the VIA-supported DOS Drivers. Enabling this setting will require system I/O, IRQ, and DMA resources. It is strongly recommended that this setting be left "Disabled."

- PCI and ISA Configuration (from the Advanced menu)

  The following setting should be retained.

  PCI IRQ Level 1-4: Autoselect for all

  PCI/PNP ISA UMB Region Exclusion: Available for all

- Power Management

  This setting is only effective under DOS. The only power management mode supported by the system is "Power-On Suspend."

- Memory Shadow

  These parameters should only be modified by advanced users. These settings can adversely affect system performance and reliability.

BIOS Console Redirection Settings

For applications where the Video interfaces is not used, the textual feedback typically sent to the monitor can be redirected to a COM port. In this manner, a system can be managed and booted without using a video connection.

The BIOS allows the following configuration options for Console Redirection to a COM port.

- COM port address: Disabled (default), COM port A, or COM port B.
  - If Console Redirection is enabled here, the associated COM port (with "A" here referring to COM 1 and "B" referring to COM 2) is enabled regardless of the COM port settings elsewhere.
- "Continue CR After POST": Off (default), or On.
  - Determines whether or not the system is to wait for a carriage return over the COM port before continuing (after POST is completed and before OS starts loading).
- Baud Rate: 19.2K (default), 300, 1200, 2400, 9600, 38.4K, 57.6K, 115.2K.
- Console Connection: Direct (default) or Modem.
- Console Type: PC ANSI (default), VT100, VT100 (8-bit), PC-ANSI (7-bit), VT100+, or VT-UTF8.
- Flow Control: CTS/RTS (default), XON-XOFF, None.
- Number of video Pages to support: 1(default) to 8.

Note: Console Redirection only works for text-based interaction. If the OS enables video and starts using direct video functions (which would be the case with a Linux X-terminal, for example), Console Redirection has no effect and video is then required.

System I/O Description

Ethernet

The Ethernet chip is the National Semiconductor DP83815 MacPhyter chip, which is connected to the system via the board's internal PCI bus.

A DOS utility program is provided for testing the chip and accessing the configuration EEPROM. Each board is factory-configured for a unique MAC address using this program. To run the program, boot the computer to DOS because the program will not run properly in a DOS window. In normal operation this program is not required.

Additional software support includes a packet driver with software to allow a full TCP/IP implementation.

Serial Ports

Helios contains four serial ports. Each port is capable of transmitting at speeds up to 115.2Kbaud. Ports COM1 and COM2 are built into the standard chipset, which are standard 16550 UARTs with 16-byte FIFOs.

Ports COM3 and COM4 are derived from an Exar 16C2850 dual UART chip and include 128-byte FIFOs. These ports may be operated at speeds to 1.5Mbaud with installation of high-speed drivers, as a custom option.

The serial ports use the following default system resources.

Port	*I/O Address Range*	*IRQ*
COM1	0x3F8 - 0x3FF	4
COM2	0x2F8 - 0x2FF	3
COM3	0x3E8 - 0x3EF	4,6,9
COM4	0x2E8 - 0x2EF	3,15

The COM1 through COM4 settings may be changed in the system BIOS. Select the *Advanced* menu, followed by *I/O Device Configuration,* to modify the base address and interrupt level.

PS/2 Ports

Helios supports two PS/2 ports.

- Keyboard
- Mouse

The PS/2 ports are accessible using a cable assembly (6981083) attached to connector J3. Support for these ports is independent of, and in addition to, mouse and keyboard support using the USB ports.

USB Ports

Four USB 2.0 ports, USB0 through USB3, are accessible using cable assemblies attached to connector J15 and J16.

USB support is intended primarily for the following devices (although any USB-standard device should function).

- Keyboard
- Mouse
- USB Floppy Drive (This is required for Crisis Recovery of boot ROM)
- USB flash disk

The BIOS supports the USB keyboard during BIOS initialization screens and legacy emulation for DOS-based applications.

The USB ports can be used for keyboard and mouse at the same time that the PS/2 keyboard and mouse are connected.

System Resources

The table below lists the default system resources utilized by the circuits on Helios.

Device	*Address*	*ISA IRQ*	*ISA DMA*
Serial Port COM1	I/O 0x3F8 – 0x3FF	4	–
Serial Port COM2	I/O 0x2F8 – 0x2FF	3	–
Serial Port COM3	I/O 0x3E8 – 0x3EF	4,6,9	–
Serial Port COM4	I/O 0x2E8 – 0x2EF	3,15	–
LPT Printer Port	I/O 0x378 – 0x37F	7	3
IDE Controller A	I/O 0x1F0 – 0x1F7	14	–
A/D Circuit (when applicable)	I/O 0x280 – 0x28F	5	–
Watchdog Timer/Serial Port/FPGA Control	I/O 0x25C-0x25F	–	–
Ethernet	OS-dependent	OS-dependent	–
USB	OS-dependent	OS-dependent	–
Sound	OS-dependent	OS-dependent	–
Video	OS-dependent	OS-dependent	–

Most of these resources are configurable and, in many cases, the Operating System alters these settings. The main devices that are subject to this dynamic configuration are on-board Ethernet, sound, video, USB, and any PC/104-*Plus* cards that are in the system. These settings may also vary depending on what other devices are present in the system. For example, adding a PC/104-*Plus* card may change the on-board Ethernet resources.

The serial port settings for COM1 and COM2 are jumper-selectable (J25, J26), whereas the settings for COM3 and COM4 are entirely software-configured in the BIOS.

Console Redirection to a Serial Port

In many applications without a local display and keyboard, it may be necessary to obtain keyboard and monitor access to the CPU for configuration, file transfer, or other operations. Helios supports this operation by enabling keyboard input and character output onto a serial port, referred to as console redirection. A serial port on another PC can be connected to the serial port on Helios with a null modem cable, and a terminal emulation program, such as HyperTerminal, can be used to establish the connection. The terminal program must be capable of transmitting special characters including F2 (some programs or configurations trap special characters).

The default Helios BIOS setting disables console redirection.

There are three possible configurations for console redirection:

- POST-only (default)
- Always On
- Disabled

To modify the console redirection settings,

1. Enter the BIOS

2. Select the Advanced menu

3. Select Console Redirection.

4. In Com Port Address, select Disabled to disable the function, On-board COM A for COM1, or On-board COM B for COM2 (default).

   If you select Disabled, you will not be able to enter BIOS again during power-up through the serial port.

To reenter BIOS when console redirection is disabled, you must either install a PC/104 video board and use a keyboard and terminal or erase the CMOS RAM, which will return the BIOS to its default settings. CMOS RAM may be erased by removing the jumper on the JP10 jumper block.

Note: Before erasing CMOS RAM, write down any custom BIOS settings you have made.

If you erase the CMOS RAM, the next time the CPU powers up COM2 returns to the default settings of 115.2Kbaud, N, 8, 1 and operates only during POST.

If you selected COMA or COMB, continue with the configuration, as follows.

1. For Console Type, select PC ANSI.

2. You can modify the baud rate and flow control here if desired.

3. At the bottom, for Continue C.R. after POST, select Off (default) to turn off after POST or select On to remain on always.

4. Exit the BIOS and save your settings.

Watchdog Timer

Helios contains a watchdog timer circuit consisting of two programmable timers, WD1 and WD2, cascaded together. The input to the circuit is WDI and the output is WDO. WDI may be triggered in hardware or in software. A special "early" version of WDO may be output on the WDO pin. When this signal is connected to WDI, the watchdog circuit is re-triggered automatically.

The duration of each timer is user-programmable. When WD1 is triggered, it begins to count down. When it reaches zero, it triggers WD2, sets WDO high, and may also generate a user-selectable combination of the following events.

- System Management interrupt (SMI)
- Hardware reset

WD2 then begins to count down. When the WD2 counter reaches zero, it unconditionally causes a hardware reset. The WD2 timer gives external circuits time to respond to the WDO event before the hardware reset occurs.

The watchdog timer circuit is programmed via I/O registers located on Page 0: Base +28-31. The Helios watchdog timer is supported in the Diamond Systems Universal Driver software version 5.7 and later.

Flash Memory

Helios contains a 512KB, 16-bit wide flash memory chip for storage of BIOS and other system configuration data.

Backup Battery

Helios contains an integrated RTC/CMOS RAM backup battery. This battery has a capacity of 120mAH and will last over three years in power-off state.

The on-board battery is activated for the first time during initial factory configuration and test. Storage temperature of the board can affect the total battery life. Storage at 23ºC is recommended.

System Reset

Helios contains a chip to control system reset operation. Reset occurs under the following conditions.

- User causes reset with a ground contact on the *Reset* input.
- Input voltage drops below 4.75V.
- Over-current condition on output power line .

The ISA Reset signal is an active high pulse with a 200ms duration. The PCI Reset is active low, with a typical pulse width duration of 200 msec.

On-Board Video

Helios provides VGA CRT and LVDS flat panel video using a Volari Z9s low-power video chip. The Volari Z9s provides VGA display output up to 1600x1200.

Data Acquisition Circuit

Helios contains a data acquisition subsystem consisting of A/D, D/A, digital I/O, and counter/timer features. This subsystem is equivalent to a complete add-on data acquisition module.

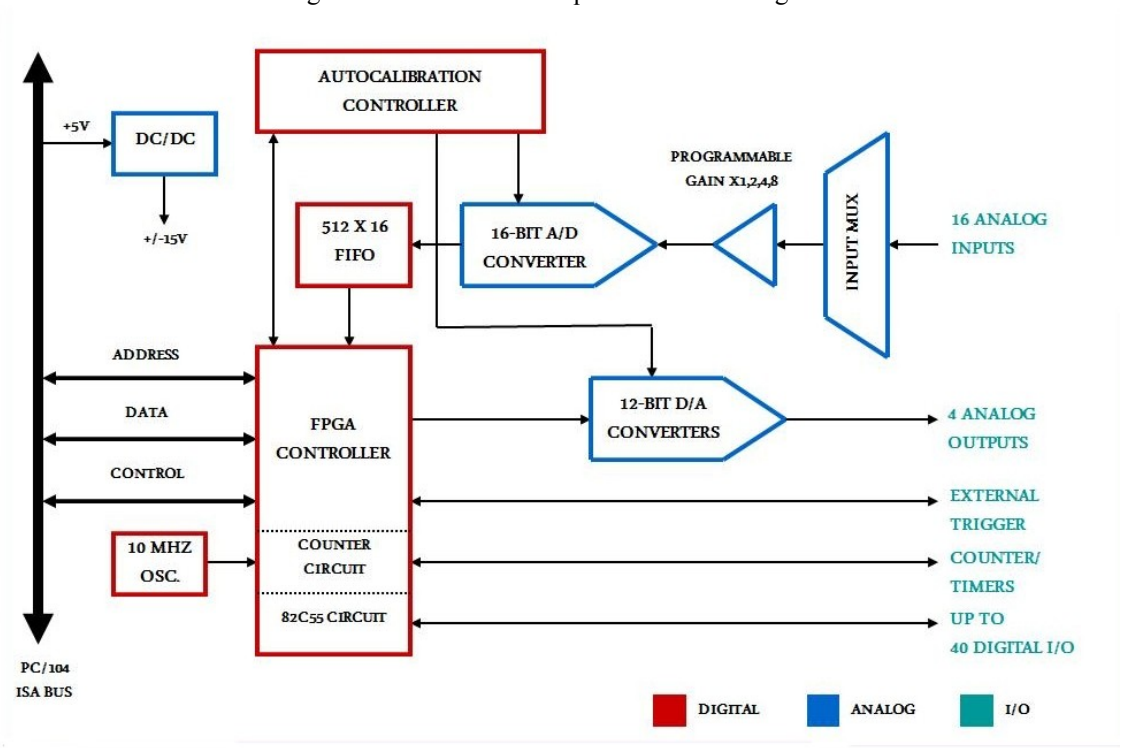
The A/D section includes a 16-bit A/D converter, 16 input channels, and a 512-sample FIFO. Input ranges are programmable, and the maximum sampling rate is 100KHz. The D/A section includes 4 12-bit D/A channels. The digital I/O section includes up to 40 lines with programmable direction. The counter/timer section includes a 24-bit counter/timer to control A/D sampling rates and a 16-bit counter/timer for user applications.

High-speed A/D sampling is supported with interrupts and a FIFO. The FIFO is used to store a user-selected number of samples, and the interrupt occurs when the FIFO reaches this threshold. Once the interrupt occurs, an interrupt routine runs and reads the data out of the FIFO. In this way the interrupt rate is reduced by a factor equal to the size of the FIFO threshold, enabling a faster A/D sampling rate. The circuit can operate at sampling rates of up to 100KHz, with an interrupt rate of 6.6-10KHz.

The A/D circuit uses the default settings of I/O address range 280h – 28Fh (base address 280) and IRQ 5. These settings can be changed if needed. The I/O address range is changed in the BIOS, and the interrupt level is changed with jumper block J21.

Figure 13 show the Helios data acquisition block diagram.

Figure 13: Helios Data Acquisition Block Diagram



Data Acquisition Circuitry I/O Map

Overview

The data acquisition circuitry on Helios occupies 16 bytes in I/O memory space. The default address range is 280h (base address) to 28Fh.

The data acquisition FPGA can be enabled/disabled in the BIOS under the Advanced menu. Scroll down to the "FPGA Mode" option and select "Enabled" or "Disabled," accordingly. If the FPGA is disabled you will not be able to interact with the data acquisition circuit. The FPGA can also be enabled or disabled programmatically through the CPLD.

Register Map Page Summary

The following table summarizes the DAC register functions. The registers are paged to allow access to enhanced functions. There are three register pages and the desired page is selected using the A/D gain and scan settings register, Base+3, bits PG0-PG1, provided the board is in enhanced mode.

Page 0		
Base +	***Write Function***	***Read Function***
0	Command	A/D LSB
1	Enhanced Mode Control	A/D MSB
2	A/D Scan Channel	A/D Scan Channel L/H
3	A/D Gain and Scan Settings/Page Control	A/D-D/A Gain and Status
4	Interrupt/Counter Control	Interrupt/Counter Control
5	FIFO Threshold	FIFO Threshold
6	DAC LSB	FIFO Current Depth/FIFO Status
7	DAC MSB + Channel No.	Interrupt and A/D Channel
8	Digital I/O Port A Output	Digital I/O Port A Input
9	Digital I/O Port B Output	Digital I/O Port B Input
10	Digital I/O Port C Output	Digital I/O Port C Input
11	D/A, Digital I/O Control	D/A, Digital I/O Control
12	Counter/Timer D7-0	Counter/Timer D7-0
13	Counter/Timer D15-8	Counter/Timer D15-8
14	Counter/Timer D23-16	Counter/Timer D23-16
15	Counter/Timer Control	FPGA Revision Code

Page 1		
Base +	***Write Function***	***Read Function***
12	EEPROM/TrimDAC data latch	EEPROM/TrimDAC data
13	EEPROM/TrimDAC address latch	EEPROM/TrimDAC address
14	EEPROM/TrimDAC control	EEPROM/TrimDAC status
15	Special features unlock register 0	Page 1 select read back check

Page 2		
Base +	***Write Function***	***Read Function***
12	Expanded FIFO depth	Expanded FIFO depth
13	AD Mode Control	AD Mode Status
14	DA Mode Control	DA Mode Status
15	DA MSB 16bit Mode	Page 2 select read back check

Note 1: Page 0, registers 0-11 are accessible when Page 1 or Page 2 are selected.

Note 2: In the following tables, blank bits are not used. Writes to a blank bit have no effect and reads from a blank bit return a value of zero.

Register Map Bit Summary

Page 0 Write Register Summary

Base +	7	6	5	4	3	2	1	0
0	STRTAD	RSTBRD	RSTDA	RSTFIFO	-	CLRT	CLRD	CLRA
1	-	-	-	-	-	-	-	-
2	H3	H2	H1	H0	L3	L2	L1	L0
3	-	-	PG1	PG0	-	SCANEN	G1	G0
4	CKSEL1	CKFRQ1	CKFRQ0	ADCLK	-	TINTE	DINTE	AINTE
5	-	-	FT5	FT4	FT3	FT2	FT1	FT0
6	DA7-DA0							
7	DACH1	DACH0	-	-	DA11	DA10	DA9	DA8
8	A7	A6	A5	A4	A3	A2	A1	A0
9	B7	B6	B5	B4	B3	B2	B1	B0
10	C7	C6	C5	C4	C3	C2	C1	C0
11	DIOCTR	DAMODE	DASIM	DIRA	DIRCH	-	DIRB	DIRCL
12	CTRD7	CTRD6	CTRD5	CTRD4	CTRD3	CTRD2	CTRD1	CTRD0
13	CTRD15	CTRD14	CTRD13	CTRD12	CTRD11	CTRD10	CTRD9	CTRD8
14	CTRD23	CTRD22	CTRD21	CTRD20	CTRD19	CTRD18	CTRD17	CTRD16
15	CTRNO	LATCH	GTDIS	GTEN	CTDIS	CTEN	LOAD	CLR

Page 0 Read Register Summary

Base +	7	6	5	4	3	2	1	0
0	AD7-AD0							
1	AD15-AD8							
2	H3	H2	H1	H0	L3	L2	L1	L0
3	ADBUSY	SE/DIFF	WAIT	DACBSY	OVF	SCANEN	G1	G0
4	CKSEL1	CKFRQ1	CKFRQ0	ADCLK	-	TINTE	DINTE	AINTE
5	FT7-FT0							
6	FD7-FD4				FD3/OVF	FD2/FF	FD1/HF	FD0/EF
7	-	TINT	DINT	AINT	ADCH3	ADCH2	ADCH1	ADCH0
8	A7	A6	A5	A4	A3	A2	A1	A0
9	B7	B6	B5	B4	B3	B2	B1	B0
10	C7	C6	C5	C4	C3	C2	C1	C0
11	DIOCTR	DAMODE	DASIM	DIRA	DIRCH	-	DIRB	DIRCL
12	CTRD7	CTRD6	CTRD5	CTRD4	CTRD3	CTRD2	CTRD1	CTRD0
13	CTRD15	CTRD14	CTRD13	CTRD12	CTRD11	CTRD10	CTRD9	CTRD8
14	CTRD23	CTRD22	CTRD21	CTRD20	CTRD19	CTRD18	CTRD17	CTRD16
15	FPGA Revision Code							

Page 1 Write Register Summary

Base +	7	6	5	4	3	2	1	0
12	TDAD7-TDAD0 – or - EMM_CD7-EMM_CD0							
13	EEM_CA7	EEM_CA6	EEM_CA5	EEM_CA4	EEM_CA3	EEM_CA2	EEM_CA1	EEM_CA0
14	EE_EN	EE_RW	RUNCAL	CALMUX	TDACEN	-	-	-
15	EEPROM Access Key Register							

Page 1 Read Register Summary

Base +	7	6	5	4	3	2	1	0
12	EEM_D7-EEM_D0							
13	EEM_CA7-EEM_CA0							
14	0	TDBUSY	EEBUSY	CALMUX	-	-	-	-
15	0xA1							

Page 2 Write Register Summary

Base +	7	6	5	4	3	2	1	0
12	-	-	-	-	-	-	-	EXFIFO
13	-	-	-	-	ADPOL	ADPOLEN	ADSD	ADSDEN
14	-	-	-	-	DAPOL	DAPOLEN	DAG1	DAG0
15	DA15-DA8							

Page 2 Read Register Summary

Base +	7	6	5	4	3	2	1	0
12	-	-	-	-	-	-	-	EXFIFO
13	-	-	-	-	ADPOL	ADPOLEN	ADSD	ADSDEN
14	-	-	-	DASIZE	DAPOL	DAPOLEN	DAG1	DAG0
15	0xA2							

Page 0 Register Definitions

Command: Base+0 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	STRTAD	RSTBRD	RSTDA	RSTFIFO	-	CLRT	CLRD	CLRA

STRTAD Start an A/D conversion (trigger the A/D) when in software-trigger mode AINTE = 0 (Base+4, bit 0). Once the program writes to this bit, the A/D conversion starts and the STS bit (base+3, bit 7) goes high. The program should then monitor STS and wait for it to go low (the value of Base+3 is less than 128 or 0x80). When STS goes, low the A/D data at Base+0 and Base+1 may be read.

When AINTE = 1 (Base+4, bit 0), the A/D cannot be triggered by writing to this bit. Instead, the A/D is triggered by a signal selected by ADCLK (Base+4 bit 5).

RSTBRD Reset the entire board excluding the D/A. Writing a 1 to this bit causes all on-board registers to be reset to 0. The effect on the digital I/O is that all ports are reset to input mode, and the logic state of their pins is determined by the pull-up/pull-down configuration setting selected by the user. All A/D, counter/timer, interrupt and DMA functions cease. However, the D/A values remain constant.

Command: Base+0 (Write)

RSTDA	Reset the four analog outputs. The analog outputs are reset to either mid-scale or zero-scale, depending on the jumper configuration selected by the user. A separate reset is provided for the D/A so that the user may reset the board if needed without affecting the circuitry connected to the analog outputs.
RSTFIFO	Reset the FIFO depth to 0. This clears the FIFO, allowing additional A/D conversions to be stored in the FIFO starting at address 0.
CLRT	Writing a 1 to this bit resets the timer interrupt request flip flop.
CLRD	Writing a 1 to this bit resets the digital I/O interrupt request flip flop.
CLRA	Writing a 1 to this bit resets the analog interrupt request flip flop.

- This register performs various functions. The register bits are not data bits but, instead, command triggers. Each function is initiated by writing a 1 to a particular bit. Writing a 1 to any bit in this register does not affect any other bit in this register. For example, to reset the FIFO, write the value 0x10 (16) to this register to write a 1 to bit 4. No other function of the register will be performed. Multiple actions can be performed, simultaneously, by writing a 1 to multiple bits, using a single write operation.
- The user's interrupt routine must write to the appropriate bit prior to exiting to reset the interrupt request flip flop, enabling future interrupts. Otherwise, the interrupt line remains high, indefinitely, and no additional interrupt requests are generated by the board.

A/D LSB: Base+0 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0

AD7-AD0 A/D LSB data. The A/D data must be read LSB first, followed by MSB.

The A/D value is derived by reading two bytes from Base + 0 and Base + 1 and applying the following formula:

A/D value = (Base+0 value) + ((Base+1 value) * 256)

The value is interpreted as a two's complement, 16-bit number ranging from –32768 to +32767. This raw A/D value is converted to the corresponding input voltage and/or the engineering units represented by that voltage by applying additional application-specific formulas. Both conversions (conversion to volts and conversion to engineering units) may be combined into a single formula for efficiency.

Enhanced Mode Control: Base+1 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	KEY							

KEY Writing 0xA5 to this register enables enhanced mode, which makes pages one and two accessible. Write 0xA6 to the register to disable enhanced mode.

A/D MSB: Base+1 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8

AD15-AD8 A/D MSB data. The A/D data must be read LSB first, followed by MSB.

The A/D value is derived by reading two bytes from Base + 0 and Base + 1 and applying the following formula:

A/D value = (Base+0 value) + ((Base+1 value) * 256)

The value is interpreted as a two's complement, 16-bit number ranging from –32768 to +32767. This raw A/D value is converted to the corresponding input voltage and/or the engineering units represented by that voltage by applying additional application-specific formulas. Both conversions (conversion to volts and conversion to engineering units) may be combined into a single formula for efficiency.

A/D Channel: Base+2 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	H3	H2	H1	H0	L3	L2	L1	L0

H3-H0 High channel of A/D channel scan range.
Ranges from 0 to 15 in single-ended mode, 0 to 7 in differential mode.

L3-L0 Low channel of A/D channel scan range.
Ranges from 0 to 15 in single-ended mode, 0 to 7 in differential mode.

- The high channel must be greater than or equal to the low channel.
- When this register is written, the current A/D channel is set to the low channel, so that the next time an A/D conversion is triggered the low channel will be sampled.
- When this register is written, the WAIT bit (Base+3, bit 5) goes high for 10 microseconds to indicate that the analog input circuit is settling. During this time, an A/D conversion should not be performed because the data will be inaccurate.
- After writing a new gain setting (Base+3), the WAIT bit is also set, and the program must monitor the bit prior to starting an A/D conversion.
- The channel and gain registers can be written to in succession without waiting for the intervening WAIT signal. Only one WAIT period must be observed between the last triggering condition (write to Base+2 or Base+3) and the start of an A/D conversion.
- The A/D circuit is designed to automatically increment the A/D channel each time a conversion is generated. This allows the user to avoid needing to write to the A/D channel each time. The A/D channel rotates through the values between LOW and HIGH. For example, if LOW = 0 and HIGH = 3, the A/D channels progresses through the following sequence: 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, ….
- Reading from this register returns the value previously written to it.

Analog Input Gain/Page Select/Scan Settings: Base+3 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	-	-	PG1	PG0	-	SCANEN	G1	G0

PG1-PG0 Page select (Page 0 - Page 2)

Standard Mode:
00 = Page 0
01 = Page 1
10 = Page 2

Reset and Enhanced Mode:
00 = Page 0
01 = Page 0
10 = Page 0

Note: When the board is in standard mode, only page 0 can be accessed. The page mode can only be set when the register map is in enhanced mode.

SCANEN Scan mode enable.
1 = Each A/D trigger causes the board to generate an A/D conversion on each channel in the range LOW – HIGH. The range is set with the channel register in Base+2.

Analog Input Gain/Page Select/Scan Settings: Base+3 (Write)

The STS bit (Base+3, bit 7) stays high during the entire scan.

0 = Each A/D trigger causes the board to generate a single A/D conversion on the current channel. The internal channel pointer increments to the next channel in the range LOW – HIGH or resets to LOW, if the current channel is HIGH.

The STS bit (Base+3, bit 7) stays high during the A/D conversion.

G1-G0 Analog input gain. The gain is the ratio of the voltage seen by the A/D converter and the voltage applied to the input pin. The gain setting is the same for all input channels.

- When this register is written, the WAIT bit (Read Base+3, bit 6) goes high for 10 microseconds to indicate that the analog input circuit is settling. During this time, an A/D conversion should not be performed because the data will be inaccurate. After writing a new gain setting, the program should monitor the WAIT bit prior to starting an A/D conversion.
- After writing a new channel selection (Base+2), the WAIT bit is also set, and the program must monitor it prior to starting an A/D conversion.
- The channel and gain registers can be written to in succession without waiting for the intervening WAIT signal. Only one WAIT period must be observed between the last triggering condition (write to Base+2 or Base+3) and the start of an A/D conversion.
- The following table lists the possible analog input ranges.

G1	*G0*	*Gain*	*Unipolar Range*	*Bipolar Range*
0	0	1	Invalid	±10V
0	1	2	0 - 10V	±5V
1	0	4	0 - 5 V	±2.5V
1	1	8	0 - 2.5V	±1.25V

Analog Input Status: Base+3 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	ADBUSY	SE/DIFF	WAIT	DACBSY	OVF	SCANEN	G1	G0

ADBUSY A/D status.

1 = A/D conversion or scan in progress.

0 = A/D is idle.

If SCANEN = 0, single conversion mode, STS goes high when an A/D conversion is started and stays high until the conversion is finished. If SCANEN = 1, scan mode enabled, STS stays high during the entire scan. After starting a conversion in software, the program must monitor STS and wait for the value to be 0 before reading A/D values from Base+0 and Base+1.

SE/DIFF Single-ended/differential mode indicator.

1 = Single-ended

0 = Differential

WAIT A/D input circuit status.

1 = A/D circuit is settling on a new value.

0 = ok to start conversion.

WAIT goes high after the channel register (Base+2) or the gain register (Base+3) changes, and remains high for nine microseconds. The program should monitor this bit after writing to either the channel or gain register, and wait for the value to become 0 prior to starting an A/D conversion.

DACBSY DAC is busy updating indicator (approx. 30 µS)

Analog Input Status: Base+3 (Read)

1 = Busy

0 = Idle

Do not attempt to write to the DAC (Base+6 and Base+7) while the value of this bit is 1.

OVF — FIFO Overflow bit. This bit indicates that the FIFO has overflowed, meaning that the A/D circuit has attempted to write data to a full FIFO. This condition occurs when data is written into the FIFO faster than the FIFO is read.

When overflow occurs, the FIFO discards additional data until it is reset. The OVF condition is sticky, with the bit remaining set until the FIFO is reset, allowing the application program to determine if overflow has occurred. If overflow occurs, then you must either reduce the sample rate or increase the efficiency of your interrupt routine and/or operating system.

SCANEN — Scan mode readback. (See Base+3, write, above).

G1-G0 — Gain. The gain is the ratio between the input voltage and the voltage seen by the A/D converter. The A/D always works with a maximum input voltage of 10V. A gain of two means the maximum input voltage at the connector pin is 5V.

0 = gain of 1

1 = gain of 2

2 = gain of 4

3 = gain of 8

(See the description for register Base+3, write, above).

Interrupt/DMA/Counter Control: Base+4 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	CKSEL1	CKFRQ1	CKFRQ0	ADCLK	-	TINTE	DINTE	AINTE

CKSEL1 Clock source selection for counter/timer 1.
0 = Internal oscillator, frequency selected by CLKFRQ1
1 = External clock input CLK1 (DIO C pins must be set for ctr/timer signals)

CLFRQ1 Input frequency selection for counter/timer 1 when CKSEL1 = 1.
0 = 10MHz
1 = 100KHz

CKFRQ0 Input frequency selection for counter/timer 0.
0 = 10MHz
1 = 1MHz

ADCLK A/D trigger select when AINTE = 1.
0 = Internal clock output from counter/timer 0
1 = External clock input EXTTRIG

TINTE Enable timer interrupts.
1 = Enable
0 = Disable

DINTE Enable digital I/O interrupts.
1 = Enable
0 = Disable

AINTE Enable analog input interrupts.
1 = Enable
0 = Disable

NOTE: When AINTE = 1, the A/D cannot be triggered by writing to Base + 0.

- Analog output interrupts are not supported on this board.
- Multiple interrupt operations may be performed, simultaneously. All interrupts are at the same interrupt level. The user's interrupt routine must monitor the status bits to know which circuit has requested service. After processing the data but before exiting, the interrupt routine must clear the appropriate interrupt request bit, using the Base+0 register.

FIFO Threshold: Base+5 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	-	-	FT5	FT4	FT3	FT2	FT1	FT0

FT0-FT5 FIFO threshold. When the number of A/D samples in the FIFO reaches this number, the board generates an interrupt and sets AINT high (Base+7, bit 4). The interrupt routine is responsible for reading the correct number of samples out of the FIFO.

The valid range is 1 to 48. A value of 48 is used, if a value greater than 48 is written to this register. A value of 1 is used, if 0 is written to this register. The interrupt rate is equal to the total sample rate divided by the FIFO threshold. Generally, for higher sampling rates a higher threshold should be used to reduce the interrupt rate. However, remember that the higher the FIFO threshold, the smaller the amount of FIFO space remaining to store data while waiting for the interrupt routine to respond. If a FIFO overflow condition occurs, lower the FIFO threshold and/or lower the A/D sampling rate.

DAC LSB: Base+6 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	DA7-DA0							

DA7-DA0 D/A LSB data.

D/A data is an unsigned 12-bit value. This register must be written to before Base+7, because writing to Base+7 immediately updates the DAC.

A/D Channel and FIFO Status: Base+6 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	FD7	FD6	FD5	FD4	FD3/OVF	FD2/FF	FD1/HF	FD0/EF

FD5-FD0 Current FIFO depth.

This value indicates the number of A/D values currently stored in the FIFO.

OVF FIFO overflow. This bit indicates that the FIFO has overflowed.

FF FIFO full.

HF FIFO half full; the FIFO is at least half full, containing at least 1K words of A/D data.

EF FIFO empty.

DAC MSB + Channel No.: Base+7 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	DACH1	DACH0	-	-	DA11	DA10	DA9	DA8

DACH0-1 D/A channel.
The values written to Base+6 and Base+7 are written to the selected channel, and that channel is immediately updated. The update takes approximately 20 microseconds because of the DAC serial interface.

DA8-DA11 D/A bits 8 to 11.
DA11 is the MSB. D/A data is an unsigned 12-bit value.

Analog Operation Status: Base+7 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	-	TINT	DINT	AINT	ADCH3	ADCH2	ADCH1	ADCH0

TINT Timer interrupt status.
1 = interrupt pending
0 = interrupt not pending

DINT Digital I/O interrupt status.
1 = interrupt pending
0 = interrupt not pending

AINT Analog input interrupt status.
1 = interrupt pending
0 = interrupt not pending

ADCH0-3 Current A/D channel. This is the channel sampled on the next conversion.

- When any of bits 7–4 are 1, the corresponding circuit is requesting service. The interrupt routine must poll these bits to determine which circuit needs service and then act accordingly.

Digital I/O Port A: Base+8 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	A7	A6	A5	A4	A3	A2	A1	A0

A0-A7 Port A data. The register direction is controlled by bits in the register Base+11, below.

Digital I/O Port B: Base+9 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	B7	B6	B5	B4	B3	B2	B1	B0

B0-B7 Port B data. The register direction is controlled by bits in the register Base+11, below.

Digital I/O Port C: Base+10 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	C7	C6	C5	C4	C3	C2	C1	C0

C0-C7 Port C data. The register direction is controlled by bits in the register Base+11, below.

D/A, Digital I/O Control: Base+11 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	DIOCTR	DAMODE	DASIM	DIRA	DIRCH	-	DIRB	DIRCL

DIOCTR Selects counter I/O signals or digital I/O lines C4-C7, on pins 21-24 of J14. If DIOCTR = 0, the pin direction is as shown in the following table. If DIOCTR = 1, the pin direction is controlled by the DIRCH bit.

Pin No.	***DIOCTR = 1***	***DIOCTR = 0***	***Pin direction, for DIOCTR = 0***
21	C4	Gate0	Input
22	C5	Gate1	Input
23	C6	Clk1	Input
24	C7	Out0	Output

This bit resets to 1.

DAMODE 16/12-bit DAC mode.

0 = Use the DAC in 12-bit operation.

1 = Use the DAC in the enhanced 16-bit mode.

Note: The standard Helios board uses a 12-bit DAC. Only custom boards have a 16-bit DAC installed. It is possible to identify which DAC is installed by reading the register base+14 bit, DASIZE.

DASIM D/A simultaneous update.

0 = The D/A channels will update as soon as the DAC MSB is written at either base+7 (12-bit mode) or base+15, page 2, (16-bit mode).

1 = D/A conversions are "latched." This means that the 12/16-bit values will be loaded into the D/A converter but the outputs will not change until the enhanced register at base+15, page 2, is read. When register base+15, page 2, is read, all latched D/A channels written to previously are updated. However, users should check that the DAC has completed writing to the channels before updating the outputs by reading DABSY.

DIRA Port A direction.

0 = output

1 = input

DIRCH Port C, bits 7-4, direction.

0 = output

1 = input

DIRB Port B direction.

0 = output

1 = input

DIRCL Port C, bits 0-3, direction.

0 = output

1 = input

Counter/Timer Bits 0-7: Base+12 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	D7	D6	D5	D4	D3	D2	D1	D0

D0-D7 LSB for counter 0 and counter 1. Counter 0 is 24 bits wide and counter 1 is 16 bits wide.

When writing to this register, an internal load register is loaded. Upon issuing a Load command, using Base+15, the selected counter's LSB register is loaded with this value.

When reading from this register, the LSB value of the most recent Latch command is returned.

Note: The value returned is NOT the value written to this register.

Counter/Timer Bits 8-15: Base+13 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	D15	D14	D13	D12	D11	D10	D9	D8

D15-D8 Used for counter 0 and counter 1. Counter 0 is 24 bits wide and counter 1 is 16 bits wide.

When writing to this register, an internal load register is loaded. Upon issuing a Load command, using Base+15, the selected counter's associated register is loaded with this value.

When reading from this register, the byte associated with the most recent Latch command is returned.

Note: The value returned is NOT the value written to this register.

Counter/Timer Bits 16-23: Base+14 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	23	22	21	20	19	18	17	16

D16-D23 This register is used for 24-bit wide Counter 0, only.

When writing to this register, an internal load register is loaded. Upon issuing a Load command, using Base+15 for Counter 0, the counter's MSB register is loaded with this value. (When issuing a Load command for counter 1, this register is ignored).

When reading from this register, the MSB value of the most recent Latch command for counter 0 is returned.

Note: The value returned is NOT the value written to this register.

Counter/Timer Control: Base+15 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	CTRNO	LATCH	GTDIS	GTEN	CTDIS	CTEN	LOAD	CLR

CTRNO Select counter number: 0 or 1.

LATCH Latch the selected counter to read its value. T he counter must be latched before it is read. Reading from registers 12-14 returns the most recently latched value. If you are reading Counter 1 data, read only Base+12 and Base+13. Any data in Base+14 is from the previous Counter 0 access.

GTDIS Disable external gating for the selected counter.

GTEN Enable external gating for the selected counter. If enabled, the associated gate signal, GATE0 or GATE1, controls counting on the counter. If the GATE*n* signal is high, counting is enabled. If the GATE*n* signal is low, counting is disabled.

CTDIS Disable counting on the selected counter. The counter ignores input pulses.

CTEN Enable counting on the selected counter. The counter decrements with each input pulse.

LOAD Load the selected counter with the data written to Base+12 through Base+14 or Base+12 and Base+13, depending on which counter is being loaded.

CLR Clear the current counter, setting its value to 0.

This register is used to control the counter/timers. A counter is selected in bit 7 followed by a 1 written to any one of bits 6 – 0, to select the desired operation for that counter. The other bits and associated functions are not affected. Only one operation can be performed at a time.

FPGA Revision Code: Base+15 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	REV7	REV6	REV5	REV4	REV3	REV2	REV1	REV0

REV0-7 Revision code, read as a two-digit hexadecimal value. For example, a value of 0x10 is revision 1.0

Page 1 Register Definitions

Trim DAC data/EEM Data: Base+12 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	TDAD7-TDAD0 - or - EEM_CD7-EEM_CD0							

TDAD7-TDAD0 TrimDAC data to set the DAC output value to at the selected address. TrimDAC data can only be written when TDABSY (base+14) is not set. TrimDAC address can be written by writing to this register or through the EEM mode, (EEM-WriteTDA_Data).

The reset value is zero.

EEM_CD7-EEM_CD0 Data for the command data for the EEPROM. Data can only be written when EEMBSY = 0 (base+14, page 1).

EEM Data: Base+12 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	EEM_D7-EEM_D0							

EEM_D7-EEM_D0 EEM data pointed to by EEPROM command address register (base+13, page 1).

EEPROM Command/Trim DAC Address: Base+13 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	EEM_CA7	EEM_CA6	EEM_CA5	EEM_CA4	EEM_CA3	EEM_CA2/ TDAA2	EEM_CA1/ TDAA1	EEM_CA0/ TDAA0

EEM_CA7-EEM_CA0 EEPROM command address when EEPROM write enable. Can only write data when EEMBSY (base+14) is cleared.

TDAA2-TDAA0 TrimDAC address, 8x8 bytes.

0: Q1,DAC1 ADCOFF range adjustment.
1: Q2,DAC2 ADCOFF fine adjustment.
2: Q3,DAC3 ADCFUL range adjustment.
3: Q4,DAC4 ADCFUL fine adjustment.
4: Q5,DAC5 DACOFF range adjustment.
5: Q6,DAC6 DACOFF fine adjustment.
6: Q7,DAC7 DACFUL range adjustment.
7: Q8,DAC8 DACFUL fine adjustment.

TrimDAC address can be written by writing to this register or through the EEM mode (). TrimDAC data can only be written when TDABSY (base+14) is not set.

Reset value is zero.

EEM Command Address: Base+13 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	EEM_CA7-EEM_CA0							

EEM_CA7-EEM_CA0 EEPROM command address.

Auto-CAL/Trim DAC: Base+14 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	EE_EN	EE_RW	RUNCAL	CALMUX	TDACEN	-	-	-

EE_EN EEPROM enable. Write a 1 to this bit to initiate EEPROM data transfer in the direction indicated by the EE_RW bit.

EE_RW EEPROM read/write operation select.

0 = write.

1 = read.

RUNCAL Write a 1 to this bit to cause the board to reload the calibration settings from the EEPROM.

CALMUX Calibration multiplexer enable.

0 = Disable the calibration multiplexer, enabling user analog input channels.

1 = Enable the calibration multiplexer, disabling user analog input channels.

Note: The calibration multiplexer is used to read precision on-board reference voltages that are used in the auto-calibration process. It can also be used to read the value of analog output 0.

TDACEN TrimDAC enable. Write a 1 to this bit to initiate a transfer to the TrimDAC, which is used in th auto-calibration process.

Trim DAC/EEM/Auto-CAL Status: Base+14 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	0	TDABSY	EEMBSY	ADCMEN	-	-	-	-

TDABSY TrimDAC busy flag.

1 = TrimDAC registers do not accept data, address or the start command.

EEMBSY EEPROM busy flag.

1 = EEPROM busy.

ADCMEN Multiplexer auto-calibrate mode.

1 = auto-calibrate enabled.

Write Enable: Base+15 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	WREN7-WREN0							

WREN7/0 EEPROM write enable.
Write the value 0xA5 before starting an EEPROM write command, after setting the PAGE bit.

Note: This register can only be written when EEMBSY (base+14) is cleared.

Write the value 0xA6 to unlock (enable) all enhanced features and set the FIFO depth to 1024 samples.

Page 1 Select Read Back Check: Base+15 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	PG1ID							

PGID Register page 1 ID. This register confirms page 1 access when the value 0xA1 is read.

Page 2 Register Definitions

ADC Expanded FIFO: Base+12 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	0	0	0	0	0	0	0	EXFIFO

EXFIFO ADC expanded FIFO mode flag.
0 = Not in expanded FIFO mode.
1 = In expanded FIFO mode.

Note: The FIFOs can be tracked with the registers at base+6.

ADC Control/Status: Base+13 (Read/Write)

Bit:	7	6	5	4	3	2	1	0
Name:	0	0	0	0	ADPOL	ADPOLEN	ADSD	ADSDEN

ADPOL Unipolar output selection. Reading this bit while ADPOLEN is set returns the last value written to ADPOL. Reading while ADPOLEN is clear returns the logical state of the pin.

0 = Bipolar operation.

1 = Unipolar operation.

ADPOLEN Enable ADPOL When this bit is set, the ADPOL setting is output to the DAQ circuit.

ADSD Single-ended/differential output selection. Reading this bit while ADSDEN is set returns the last value written to ADSD. Reading this bit while ADSDEN is clear returns the logical state of the pin.

0 = Differential operation.

1 = Single-ended operation.

ADSDEN Enable ADSD. When this bit is set, the ADSD setting is output to the DAQ circuit.

DA Mode Control: Base+14 (Write)

Bit:	7	6	5	4	3	2	1	0
Name:	-	-	-	-	DAPOL	DAPOLEN	DAG1	DAG0

DAPOL Unipolar output setting. Reading this bit while DAPOLEN is set returns the last value written to DAPOL. Reading this bit while DAPOLEN is clear returns the logical state of the pin.
0 = Bipolar operation.
1 = Unipolar operation.

DAPOLEN Enable DAPOL. When this bit is set, the DAPOL setting is output to the DAQ circuit.

DAG1-DAG0 D/A converter output voltage gain.

DAG1	***DAG0***	***DAPOL***	***Description***
0	0	0	5V span (0 to +5V, unipolar).
0	0	1	5V span (-2.5V to +2.5V, bipolar).
0	1	0	10V span (0 to +10V, unipolar).
0	1	1	10V span (=5V to +5V, bipolar).
1	0	0	Not used (sets output to 0V).
1	0	1	20V span (-10V to +10V, only).
1	1	0	Not used (sets output to 0V).
1	1	1	D/A converter shutdown.

Note: This register functions as a DA control jumper override for the DAQ subsection. On reset, the values default to zero.

DA Mode Status: Base+14 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	-	-	-	DASIZE	DAPOL	DAPOLEN	DAG1	DAG0

DASIZE DAC configuration.
0 = The board is equipped with a 12-bit DAC, which is the standard configuration.
1 = The board is equipped with a 16-bit DAC.

DAPOL Unipolar output setting (See base+14 write register description).

DAPOLEN Enable DAPOL setting (See base+14 write register description).

DAG1-DAG0 D/A converter output voltage gain setting (See base+14 write register description).

Select Read Back Check: Base+15 (Read)

Bit:	7	6	5	4	3	2	1	0
Name:	PG2ID							

PGID Register page 2 ID. This register confirms page 2 access when the value 0xA2 is read.

Analog-to-Digital Input Ranges and Resolution

Overview

Helios uses a 16-bit A/D converter. The full range of numerical values for a 16-bit number is 0 - 65535. However, the A/D converter uses two's complement notation, so the A/D value is interpreted as a signed integer, ranging from –32768 to +32767.

The smallest change in input voltage that can be detected is 1/(216), or 1/65536, of the full-scale input range. This smallest change results in an increase or decrease of 1 in the A/D code, and is referred to as 1 LSB (1 Least Significant Bit).

The analog inputs on Helios have three configuration options.

- Single-ended or differential mode
- Unipolar or bipolar mode
- Input range (gain)

The single-ended/differential and unipolar/bipolar modes are configured using jumper block J13, and apply to all inputs. The input range selection is done in software.

Input Range Selection

You can select a gain setting for the inputs, which causes them to be amplified before they reach the A/D converter. The gain setting is controlled in software, which allows it to be changed on a channel-by-channel basis. In general, you should select the highest gain (smallest input range) that allows the A/D converter to read the full range of voltages over which the input signals will vary. However, a gain that is too high causes the A/D converter to clip at either the high end or low end, and you will not be able to read the full range of voltages on your input signals.

Input Range Table

The table below indicates the analog input range for each possible configuration. The polarity is set using jumper block J13, and the gain is set with the G1 and G0 bits in the register at Base+3. The Gain value in the table is provided for clarity. Note that the single-ended vs. differential setting has no impact on the input range or the resolution.

Polarity	*G1*	*G0*	*Input Range*	*Resolution 1LSB*
Bipolar	0	0	±10V	305µV
Bipolar	0	1	±5V	153µV
Bipolar	1	0	±2.5V	76µV
Bipolar	1	1	±1.25V	38µV
Unipolar	0	0	Invalid	Invalid
Unipolar	0	1	0 - 10V	153µV
Unipolar	1	0	0 - 5V	76µV
Unipolar	1	1	0 - 2.5V	38µV

Performing an A/D Conversion

Introduction

This chapter describes the steps involved in performing an A/D conversion on a selected input channel using direct programming (without the driver software). Performing an A/D conversion according to the following steps. Each step is discussed in detail, below.

1. Select the input channel.

2. Select the input range.

3. Wait for analog input circuit to settle.

4. Initiate an A/D conversion.

5. Wait for the conversion to finish.

6. Read the data from the board.

7. Convert the numerical data to a meaningful value.

Select the Input Channel

To select the input channel to read, write a low-channel/high-channel pair to the channel register at Base+2. The low four bits select the low channel, and the high four bits select the high channel. When you write any value to this register, the current A/D channel is set to the low channel.

For example, to set the board to channel 4 only, write 0x44 to Base+2). To set the board to read channels 0 through 15, write 0xF0 to Base+2.

When you perform an A/D conversion, the current channel automatically increments to the next channel in the selected range. Therefore, to perform A/D conversions on a group of consecutively-numbered channels, you do not need to write the input channel prior to each conversion. For example, to read from channels 0 - 2, write 0x20 to base+2. The first conversion is on channel 0, the second will be on channel 1 and the third will be on channel 2. The channel counter wraps around to the beginning so the fourth conversion will be on channel 0, again.

If you are sampling the same channel repeatedly, set both high and low to the same value as in the first example, above. On subsequent conversions, you do not need to set the channel again.

Select the Input Range

Select the input range from among the available ranges. If the range is the same as for the previous A/D conversion it does not need to be set again. Write this value to the input range register at Base+3.

For example, for ±5V range (gain of 2), write 0x01 to Base+3.

Wait for Analog Input Circuit to Settle

After writing to either the channel register, Base+2, or the input range register, Base+3, allow time for the analog input circuit to settle before starting an A/D conversion. The board has a built-in 10µS timer to assist with the wait period. Monitor the WAIT bit at Base+3, bit 5. When the bit value is 1, the circuit is actively settling on the input signal. When the value is 0, the board is ready to perform A/D conversions.

Perform an A/D Conversion on the Current Channel

After the above steps are completed, start the A/D conversion by writing to Base+0. This write operation only triggers the A/D if AINTE = 0 (interrupts are disabled). When AINTE = 1, the A/D can only be triggered by the on-

board counter/timer or an external signal. This protects against accidental triggering by software during a long-running interrupt-based acquisition process.

```
outp(base,0x80);
```

Wait for the Conversion to Finish

The A/D converter chip takes up to five microseconds to complete one A/D conversion. Most processors and software can operate fast enough so that if you try to read the A/D converter immediately after starting the conversion, the read will occur faster than the A/D conversion and return invalid data. Therefore, the A/D converter provides a status signal to indicate whether it is busy or idle. This bit can be read back from the status register at Base+3, bit 7. When the A/D converter is busy (performing an A/D conversion), the bit value is 1 and the program must wait. When the A/D converter is idle (conversion is done and data is available), this bit value is 0 and the program may read the data.

The following statement is a simple example of this operation.

```
while (inp(base+3) & 0x80);  // Wait for conversion to finish before proceeding
```

The above example could hang your program if there is a hardware fault and the bit is stuck at 1. A better solution is to use a loop with a timeout, as shown below.

```
int checkstatus()    // returns 0 if ok, -1 if error
int i;
for (i = 0; i < 10000; i++)
{
    if !(inp(base+3) & 0x80) then return(0);    // conversion completed
}
return(-1);                                     // conversion did not complete
```

Read the Data from the Board

Once the conversion is complete, you can read the data back from the A/D converter. The data is a 16-bit value and is read back in two 8-bit bytes. The LSB must be read from the board before the MSB because the data is inserted into the board's FIFO in that order. Unlike other registers on the board, the A/D data may only be read one time, because each time a byte is read from the FIFO the internal FIFO pointer advances and that byte is no longer available. Reading data from an empty FIFO returns unpredictable results.

The following pseudo-code illustrates how to read and construct the 16-bit A/D value.

```
LSB = inp(base);
MSB = inp(base+1);
Data = MSB * 256 + LSB;   // combine the 2 bytes into a 16-bit value
```

The final data are interpreted as a 16-bit signed integer in the range -32768 to +32767.

> **Note: The data range always includes both positive and negative values, even if the board is set to a unipolar input range. The data must now be converted to volts or other engineering units by using a conversion formula, as discussed below.**

In scan mode, the behavior is the same except when the program initiates a conversion, all channels in the programmed channel range will be sampled once and the data will be stored in the FIFO. The FIFO depth register increments by the scan size. When STS goes low, the program should read out the data for all channels.

Convert the numerical data to a meaningful value

Once the A/D value is read, it needs to be converted to a meaningful value. The first step is to convert it back to the actual measured voltage. Afterwards, you may need to convert the voltage to some other engineering units. For

example, the voltage may come from a temperature sensor and the voltage would then need to be converted to the corresponding temperature, according to the temperature sensor's characteristics.

Since there are a large number of possible input devices, this secondary step is not included here. Only conversion to input voltage is described. However, you can combine both transformations into a single formula if desired.

To convert the A/D value to the corresponding input voltage, use the following formulas.

Conversion Formula for Bipolar Input Ranges

Input voltage = A/D value / 32768 * Full-scale input range

Example:

Given, Input range is ±5V and A/D value is 17761.

Therefore,

Input voltage = 17761 / 32768 * 5V = 2.710V.

For a bipolar input range,

1 LSB = 1/32768 * Full-scale voltage.

The table, below, shows the relationship between A/D code and input voltage for a bipolar input range (VFS = Full scale input voltage).

A/D Code	***Input Voltage Symbolic Formula***	***Input Voltage for ±5V Range***
-32768	$-V_{FS}$	-5.0000V
-32767	$-V_{FS}$ + 1 LSB	-4.9998V
...	...	...
-1	-1 LSB	-0.00015V
0	0	0.0000V
1	+1 LSB	0.00015V
...	...	...
32767	V_{FS} - 1 LSB	4.9998V

Conversion Formula for Unipolar Input Ranges

Input voltage = (A/D value + 32768) / 65536 * Full-scale input range

Example:

Given, Input range is 0-5V and A/D value is 17761.

Therefore,

Input voltage = (17761 + 32768) / 65536 * 5V = 3.855V.

For a unipolar input range, 1 LSB = 1/65536 * Full-scale voltage.

The following table illustrates the relationship between A/D code and input voltage for a unipolar input range (VFS = Full scale input voltage).

A/D Code	*Input Voltage Symbolic Formula*	*Input Voltage for 0–5V Range*
-32768	0V	0.0000V
-32767	1 LSB (V_{FS} / 65536)	0.000076V
...	...	...
-1	V_{FS} / 2 - 1 LSB	2.4999V
0	V_{FS} / 2	2.5000V
1	V_{FS} / 2 + 1 LSB	2.5001V
...	...	...
32767	V_{FS} - 1 LSB	4.9999V

A/D Scan, Interrupt and FIFO Operation

The control bits SCANEN (scan enable) and AINTE (A/D interrupt enable) in conjunction with the FIFO determine the behavior of the board during A/D conversions and interrupts.

At the end of an AD conversion, the 16-bit A/D data is latched into the 8-bit FIFO in an interleaved fashion, first LSB, then MSB. A/D Data is read out of the FIFO with 2 read operations, first Base + 0 (LSB) and then Base + 1 (MSB).

When SCANEN = 1, each time an A/D trigger occurs, the board will perform an A/D conversion on all channels in the channel range programmed in Base + 2. When SCANEN = 0, each time an A/D trigger occurs, the board will perform a single A/D conversion and then advance to the next channel and wait for the next trigger.

During interrupt operation (AINTE = 1), the FIFO will fill up with data until it reaches the threshold programmed in the FIFO threshold register, and then the interrupt request will occur. If AINTE = 0, the FIFO threshold is ignored and the FIFO continues to fill up.

If the FIFO reaches its limit of 48 samples, then the next time an A/D conversion occurs the Overflow flag OVF will be set. In this case the FIFO will not accept any more data, and its contents will be preserved and may be read out. In order to clear the overflow condition, the program must reset the FIFO by writing to the RSTFIFO bit in Base + 1, or a hardware reset must occur.

In Scan mode (SCANEN = 1), the FIFO threshold should be set to a number at least equal to the scan size and in all cases equal to an integral number of scans. For example if the scan size is 8 channels, the FIFO threshold should be set to 8, 16, 24, 32, 40, or 48, but not less than 8. This way the interrupt will occur at the end of the scan, and the interrupt routine can read in a complete scan or set of scans each time it runs.

In non-scan mode (SCANEN = 0), the FIFO threshold should be set to a level that minimizes the interrupt rate but leaves enough time for the interrupt routine to respond before the next A/D conversion occurs. Remember that no data is available until the interrupt occurs, so if the rate is slow the delay to receive A/D data may be long. Therefore for slow sample rates the FIFO threshold should be small. If the sample rate is high, the FIFO threshold should be high to reduce the interrupt rate. However remember that the remaining space in the FIFO determines the time the interrupt routine has to respond to the interrupt request. If the FIFO threshold is too high, the FIFO may overflow before the interrupt routine responds. A good rule of thumb is to limit the interrupt rate to no more than 1,000-2,000 per second in Windows and Linux or 10,000 per second in DOS. Experimentation may be necessary to determine the optimum FIFO threshold for each application.

The table on the next page describes the board's behavior for each of the 4 possible cases of AINTE and SCANEN. The given interrupt software behavior describes the operation of the Diamond Systems Universal Driver software. If you write your own software or interrupt routine you should conform to the described behavior for optimum results.

The following table describes the register settings for the A/D operating modes. (LOW and HIGH channels referenced in the table are the 4-bit channel numbers in Base+2.)

AINTE Base+4 bit 0	*SCANE Base+2 bit 1*	*Operation*
0	0	Single A/D conversions are triggered by write to B+0. STS stays high during the A/D conversion. No interrupt occurs. The user program monitors STS (Base+3, bit 7) and reads A/D data when STS goes low.
0	1	A/D scans are triggered by write to B+0. All channels between LOW and HIGH are sampled. STS stays high during the entire scan (multiple A/D conversions). No interrupt occurs. The user program monitors STS (Base+3, bit 7) and reads A/D data when STS goes low.
1	0	Single A/D conversions are triggered by the source selected with ADCLK (Base+4, bit 4). STS stays high during the A/D conversion. A/D interrupt occurs when the FIFO reaches its programmed threshold. The interrupt routine reads the number of samples equal to the FIFO threshold (Base+5, bits 0-5).
1	1	A/D scans are triggered by the source selected with ADCLK (Base+4, bit 4). STS stays high during the entire scan (multiple A/D conversions). A/D interrupt occurs when the FIFO reaches its programmed threshold. The interrupt routine reads the number of samples equal to the FIFO threshold (Base+5, bits 0-5).

Digital-to-Analog Output Ranges and Resolution

Description

Helios uses a 4-channel 12-bit D/A converter (DAC) to provide four analog outputs. A 12-bit DAC can generate output voltages with the precision of a 12-bit binary number. The maximum value of a 12-bit binary number is 2^{12} - 1, or 4095, so the full range of numerical values that the DACs support is 0 - 4095. The value 0 always corresponds to the lowest voltage in the output range, and the value 4095 always corresponds to the highest voltage minus 1 LSB. The theoretical top end of the range corresponds to an output code of 4096 which is impossible to achieve.

> **Note: In this manual, the terms analog output, D/A, and DAC are all used interchangeably to mean the conversion of digital data originating from the Helios computer hardware to an analog signal terminating at an external source.**

Resolution

The resolution is the smallest possible change in output voltage. For a 12-bit DAC the resolution is $1/(2^{12})$, or 1/4096, of the full-scale output range. This smallest change results from an increase or decrease of 1 in the D/A code, so this change is referred to as 1 least significant bit (1 LSB). The value of this LSB is calculated as follows.

1 LSB = Output voltage range / 4096

Example:

For, Output range = 0-10V,

Output voltage range = 10V – 0V = 10V

Therefore,

1 LSB = 10V / 4096 = 2.44mV

Example:

For, Output range = ±10V;

Output voltage range = 10V – (-10V) = 20V

Therefore,

1 LSB = 20V / 4096 = 4.88mV

Output Range Selection

Jumper block J23 is used to select the DAC output range. The DACs can be configured for 0-10V or ±10V.

Two parameters are configured

- unipolar/bipolar mode
- power-up/reset clear mode.

In most case, for unipolar mode configure the board to reset to zero scale, and for bipolar mode configure the board for reset to mid-scale. In each case the DACs reset to 0V.

D/A Conversion Formulas and Tables

The formulas below explain how to convert between D/A codes and output voltages.

D/A Conversion Formulas for Unipolar Output Ranges

Output voltage = (D/A code / 4096) * Reference voltage

D/A code = (Output voltage / Reference voltage) * 4096

Example:

For,

Output range in unipolar mode = 0 – 10V,

and,

Full-scale range = 10V – 0V = 10V,

if,

Desired output voltage = 2.000V,

D/A code = 2.000V / 10V * 4096 = 819.2 => 819

Note: the output code is always an integer.

For the unipolar output range 0-10V, 1 LSB = 1/4096 * 10V = 2.44mV.

The following table illustrates the relationship between D/A code and output voltage for a unipolar output range (VREF = Reference voltage).

D/A Code	***Output Voltage Symbolic Formula***	***Output Voltage for 0-10V Range***
0	0V	0.0000V
1	1 LSB (V_{REF} / 4096)	0.0024V
...	...	...
2047	V_{REF} / 2 - 1 LSB	4.9976V
2048	V_{REF} / 2	5.0000V
2049	V_{REF} / 2 + 1 LSB	5.0024V
...	...	...
4095	V_{REF} - 1 LSB	9.9976V

D/A Conversion Formulas for Bipolar Output Ranges

Output voltage = ((D/A code – 2048) / 2048) * Output reference

D/A code = (Output voltage / Output reference) * 2048 + 2048

Example:

For,

Output range in bipolar mode = ±10V

and,

Full-scale range = 10V – (-10V) = 20V

if,

Desired output voltage = 2.000V

D/A code = 2V / 10V * 2048 + 2048 = 2457.6 => 2458

For the bipolar output range ±10V, 1 LSB = 1/4096 * 20V, or 4.88mV.

The following table illustrates the relationship between D/A code and output voltage for a bipolar output range (VREF = Reference voltage).

D/A Code	***Output Voltage Symbolic Formula***	***Output Voltage for ±10V Range***
0	$-V_{REF}$	-10.0000V
1	V_{REF} + 1 LSB	-9.9951V
...	...	...
2047	-1 LSB	-0.0049V
2048	0	0.0000V
2049	+1 LSB	0.0049V
...	...	...
4095	V_{REF} - 1 LSB	9.9951V

Generating an Analog Output

There are three steps involved in performing a D/A conversion, or generating an analog output. Each step is described in more detail, below. The descriptions use direct programming instead of driver software.

1. Compute the D/A code for the desired output voltage.

2. Write the value to the selected output channel.

3. Wait for the D/A to update.

Compute the D/A Code for the Desired Output Voltage

Use the formulas in the preceding section to compute the D/A code required to generate the desired voltage.

> **Note: The DAC cannot generate the actual full-scale reference voltage; to do so would require an output code of 4096, which is not possible with a 12-bit number. The maximum output value is 4095. Therefore, the maximum possible output voltage is always 1 LSB less than the full-scale reference voltage.**

Write the Value to the Selected Output Channel Registers

Use the following formulas to compute the LSB and MSB values.

LSB = D/A Code & 255 ;keep only the low 8 bits

MSB = int(D/A code / 256) ;strip off low 8 bits, keep 4 high bits

Example:

For,

Output code = 1776

Compute,

LSB = 1776 & 255 = 240 (0xF0)

and

MSB = int(1776 / 256) = int(6.9375) = 6

The LSB is an 8-bit number in the range 0-255. The MSB is a 4-bit number in the range 0-15.

The MSB is always rounded *down*. The truncated portion is accounted for by the LSB.

Write these values to the selected channel. The LSB is written to Base+6. The MSB and channel number are written to Base+7 (MSB = bits 0-3, channel number,0-3 = bits 6-7).

```
outp(Base+6, LSB);
outp(Base+7, MSB + channel << 6);
```

Wait for the D/A to Update

Writing the MSB and channel number to Base+7 starts the D/A update process for the selected channel. The update process requires approximately 30 microseconds to transmit the data serially to the D/A chip and update the D/A

circuit in the chip. During this period, no attempt should be made to write to any other channel in the D/A through addresses Base+6 or Base+7.

The status bit DACBSY (Base+3, bit 4) indicates if the D/A is busy updating (1) or idle (0). After writing to the D/A, monitor DACBSY until it is zero before continuing with the next D/A operation.

Analog Circuit Calibration

Calibration applies only to boards with the analog I/O circuitry.

The Helios auto-calibration circuit uses an octal 8-bit TrimDAC IC to provide small adjustments to the offset and gain at various points in the circuit. Four of the DACs are used for the A/D calibration and the other four are used for the D/A. The 8-bit TrimDAC values are stored in an on-board EEPROM and are recalled automatically on power-up.

An on-board ultra-stable +5V reference chip with 5ppm offset drift is used as the voltage reference for all calibration operations. From this reference, several intermediate values are derived that are used for the calibration. One is just under +5V and one is just above 0V. These values are measured at the factory and their values are stored in the on-board EEPROM for use by the calibration program. Note that the actual values of the reference signals does not matter as long as they are stable, because the calibration routine knows the values and can adjust the calibration circuit to achieve them. An extra input multiplexor chip is used to feed the calibration voltages into the A/D circuit during the process.

For bipolar A/D calibration, first 0V is measured and the TrimDAC is adjusted until the target A/D reading is achieved. For unipolar calibration, the voltage just above 0V is used as the first measurement value. Two TrimDAC channels are used for the offset. The first channel provides a coarse adjustment ot bring the A/D readings into range and then the second channel provides a fine adjustment for maximum accuracy. The use of both coarse and fine adjustments provides a wider range of total adjustment capability. The range of the fine adjustment exceeds the smallest change in the coarse adjustment so there is no gap in the adjustment range.

After the offset is adjusted, the full-scale is adjusted in a similar manner. The reference value just under 5V is fed into the A/D and two additional TrimDACs provide coarse and fine adjustments to achieve the target A/D near-full-scale reading.

Once the A/D is completely calibrated, the 16-bit (or 12-bit) D/A channels can be calibrated. Unlike the A/D circuit, which uses a single A/D for all input channels, the D/A circuit actually contains a single D/A converter for each of the four output channels. These channels are fed into the calibration multiplexor and the remaining four TrimDAC channels are used to calibrate them in a similar manner to the A/D. A single adjustment is used for the high reference and both coarse and fine adjustments are used for the low reference.

The entire process takes about one second for each input range. Once it is complete, the board is ready to run. All eight TrimDAC values are stored in the EEPROM so that the next time power is cycled to the board the values will be loaded automatically.

Digital I/O Operation

The Helios board has 5 DIO ports.

Ports 1 through 3 (A, B, C) are controlled by the FPGA registers. These ports can be configured as either input or output. The configuration byte is located at register Base+11, and are available on the panel board on connector J4.

Ports 4 and 5 (D, E) are provided as direct ports from the processor at the following addresses:

Port	*Address*
4	0x78
5	0x79

These ports are also configurable as either input or output ports. The direction can also be set in the BIOS as described in the BIOS Setup section of this document. These ports are available on the Helios panel board on DB25 connector J1.

> **Note: All Helios DIO ports are only BYTE configurable, meaning that individual bits of the ports cannot be configured as input or output, separately. When attempting to configure a port, the entire port will be configured for either input operation or output operation unless the configuration is changed, again.**

Programming the Watchdog Timers Using the DSCUD API

The DIO functionality of the DSUCD API hides the addressing details from the application when accessing the DIO ports. The following DIO functions are available:

Function Name	*Description*
dscDIOSetConfig	Sets the configuration of the DIO ports.
dscDIOOutputByte	Sends a byte to the specified DIO port.
dscDIOInputByte	Receives a byte from the specified DIO port.
dscDIOInputBit	Receives a bit from the specified DIO port.
dscDIOOutputBit	Sets/resets a bit on the specified DIO port.

dscDIOSetConfig Function

This function sets the configuration of the DIO ports

Synopsis

```
dscDIOSetConfig ( DSCB board, BYTE *config );
```

Parameters

Name	*Data Type*	*Description*
board	*DSCB*	Board pointer from previous call to function *dscInitBoard()*.
config	*BYTE **	This is a two-byte array that the application **must** allocate to configure the individual DIO port directions in DSCUD. The following *config* array values are used to configure the FPGA DIO ports. To configure ports 1, 2 and 3 (A, B, C): *config[0]* = 0x00 *config[1]* = 0x## ; configuration byte. Refer to register Base+11. **Note: When a bit for a corresponding port is set to 0, the port operates as an output port, otherwise, the port is configured as an input port.** To configure ports 4 and 5 (D, E): Configure Port 4: *config[0]* = 0x01 *config[1]* = 0xFF ; setup the port as an input port = 0x00 ; setup the port as an output port. Configure Port 5: *config[0]* = 0x02 *config[1]* = 0xFF ; setup the port as an input port = 0x00 ; setup the port as an output port. **Note: Only values in *config[0]* are validated by DSCUD. The only valid values are 0, 1 and 2. All other values cause an Invalid Port ID error to be returned. The *config[1]* values are not validated by DSCUD and it is the responsibility of the application to assign the correct values.** The following table shows a few examples of different values that the config array can have, and the corresponding port configuration:

config[0]	*config[1]*	*Port Configuration*
0x00	0x00	PORTA = O PORTB = O PORT C = O
0x00	0x0A	PORTA = O PORT B = I PORTCH= I PORT CL = O
0x01	0xFF	PORTD = I
0x02	0x00	PORTD = O

dscDIOOutputByte Function

This function sends a byte to the DIO port specified in the port parameter.

Synopsis

```
dscDIOOutputByte ( DSCB board, BYTE port, BYTE digital_value );
```

Parameters

Name	*Data Type*	*Description*
board	*DSCB*	Board pointer from previous call to function *dscInitBoard()*.
port	*BYTE*	FPGA or CPU port specification. 0x00 = FPGA PORTA (A) 0x01 = FPGA PORTB (B) 0x02 = FPGA PORTC (C) 0x03 = CPU PORTA (D) 0x04 = CPU PORTB (E)
digital_value	*BYTE*	The value contained in the parameter named digital_value will be output to the port specified in the port parameter of the function.

dscDIOInputByte Function

This function receives a byte from the DIO port specified in the port parameter.

Synopsis

```
dscDIOInputByte ( DSCB board, BYTE port, BYTE *digital_value );
```

Parameters

Name	*Data Type*	*Description*
board	*DSCB*	Board pointer from previous call to function *dscInitBoard()*.
port	*BYTE*	FPGA or CPU port specification. 0x00 = FPGA PORTA (A) 0x01 = FPGA PORTB (B) 0x02 = FPGA PORTC (C) 0x03 = CPU PORTA (D) 0x04 = CPU PORTB (E)
digital_value	*BYTE* *	A pointer to the value read from the port specified.

dscDIOInputBit Function

This function receives a bit from the DIO port specified in the port parameter.

Synopsis

```
dscDIOInputBit ( DSCB board, BYTE port, BYTE bit, BYTE *digital_value );
```

Parameters

Name	*Data Type*	*Description*
board	*DSCB*	Board pointer from previous call to function *dscInitBoard()*.
port	*BYTE*	FPGA or CPU port specification. 0x00 = FPGA PORTA (A) 0x01 = FPGA PORTB (B) 0x02 = FPGA PORTC (C) 0x03 = CPU PORTA (D) 0x04 = CPU PORTB (E)
bit	*BYTE*	The bit to read.
digital_value	*BYTE* *	The location to return the BYTE representation of the specified *bit*. 0 = Bit value is zero. 1 = Bit value is one.

dscDIOOutputBit Function

This function sets/resets a bit on the DIO port specified in the port parameter. All the functionality is similar to dscDIOInputByte function. All the functionality is similar to dscDIOOutputByte function.

Synopsis

```
dscDIOOutputBit ( DSCB board, BYTE port, BYTE bit, BYTE digital_value );
```

Parameters

Name	*Data Type*	*Description*
board	*DSCB*	Board pointer from previous call to function *dscInitBoard()*.
port	*BYTE*	FPGA or CPU port specification. 0x00 = FPGA PORTA (A) 0x01 = FPGA PORTB (B) 0x02 = FPGA PORTC (C) 0x03 = CPU PORTA (D) 0x04 = CPU PORTB (E)
bit	*BYTE*	The bit to modify.
digital_value	*BYTE*	Value to which the specified *bit* will be set. 0 = Bit value set to zero. 1 = Bit value is set to one.

Counter/Timer Operation

Helios contains two counter/timers that provide various timing functions on the board for A/D timing and user functions. These counters are controlled with registers in the on-board data acquisition controller FPGA.

Counter 0 – A/D Sample Control

Counter 0 is a 24-bit, "divide-by-n" counter used for controlling A/D sampling. The counter has a clock input, a gate input, and an output. The input is a 10MHz or 1MHz clock provided on the board and selected with bit CKFRQ0 in register Base+4, bit 5. The gate is an optional signal that can be input on pin 21 of I/O header J14 when DIOCTR (Base+11, bit 7) is 1. If this signal is not used, the counter runs freely. The output is a positive pulse whose frequency is equal to the input clock divided by the 24-bit divisor programmed into the counter. The output appears on pin 24 of the I/O header when DIOCTR is 1.

The counter operates by counting down from the programmed divisor value. When the counter reaches zero, it outputs a positive-going pulse equal to one input clock period (100ns or 1µs, depending on the input clock selected by CKFRQ0). The counter then reloads to the initial load value and repeats the process, indefinitely.

The output frequency can range from 5MHz (10MHz clock, divisor = 2) to 0.06Hz (1MHz clock divided by 16,777,215, or 224-1). The output is fed into the A/D timing circuit and can be selected to trigger A/D conversions when Base+4 register bits AINTE is 1 and ADCLK is 0. Using the control register at Base+15, the counter can be loaded, cleared, enabled and disabled. The optional gate can be enabled and disabled, and the counter value can be latched for reading.

Counter 1 – Counting/Totalizing Functions

Counter 1 is similar to Counter 0 except that it is a 16-bit counter. Counter 1 also has an input, a gate and an output. These signals may be user-provided on the I/O header when DIOCTR is 0, or the input may come from the on-board clock generator. When the on-board clock generator is used, the clock frequency is either 10MHz or 100KHz, as determined by control Base+4 register bit CKFRQ1.

The output is a positive-going pulse that appears on pin 26 of the I/O header. The output pulse occurs when the counter reaches zero. When the counter reaches zero, it reloads and restarts on the next clock pulse. The output stays high for the entire time the counter is at zero; i.e., from the input pulse that causes the counter to reach zero until the input pulse that causes the counter to reload.

When DIOCTR is 0, Counter 1 operates as follows.

- It counts positive edges of the signal on pin 23 on the I/O header.
- The gate is provided on pin 22. If the signal is high, the counter counts. If the signal is low, the counter holds its value and ignore input pulses. This pin has a pull-up so the counter can operate without any external gate signal.

> **NOTE: When counting external pulses, Counter 1 only updates its read register every fourth pulse. This behavior is due to the synchronous design of the counter having to contend with the asynchronous input pulses. The count register contents are correct on the fourth pulse but remain static until four additional pulses occur on the input.**

When DIOCTR is 1, Counter 1 operates as follows.

- The counter takes its input from the on-board clock generator based on the value of the Base+4 register CKFRQ1 bit. There is no gating and the counter runs continuously.

Counter 1 may be used as either a pulse generator or a totalizer/counter. In pulse generator mode, the output signal on pin 26 is of interest. In totalizer/counter mode, the counter value is of interest and may be read by first latching the value and then reading it. The width of the pulse is equal to the time period of the selected counters clock source.

Command Sequences

Diamond Systems provides driver software to control the counter/timers on Helios. The information in this section is intended as a guide for programmers writing their own code, instead of using the driver, and to give a better understanding of the counter/timer operation.

The counter control register is located at I/O address base+15.

Load and Enable (Run) a Counter Sequence

1. Write the data to the counter. For counter 0, three bytes are required to load a 24-bit value. For counter 1, two bytes are needed for a 16-bit value. The value is an unsigned integer.

   Break the load value into 3 bytes: low, middle, and high, (Two bytes for Counter 1) and write the bytes to the data registers in any sequence.

   Counter 0:

   ```
   outp(base+12,low);
   outp(base+13,middle);
   outp(base+14,high);
   ```

   Counter 1:

   ```
   outp(base+12,low);
   outp(base+13,high);
   ```

2. Load the counter.

   Counter 0:

   ```
   outp(base+15,0x02);
   ```

   Counter 1:

   ```
   outp(base+15,0x82);
   ```

3. Enable the gate if desired. The gating may be enabled or disabled at any time. When gating is disabled, the counter counts all incoming edges. When gating is enabled, if the gate is high the counter counts all incoming edges and, if the gate is low, the counter ignores incoming clock edges.

   Counter 0:

   ```
   outp(base+15,0x10);
   ```

   Counter 1:

   ```
   outp(base+15,0x90);
   ```

4. Enable the counter. A counter may be enabled or disabled at any time. If disabled, the counter ignores incoming clock edges.

   Counter 0:

   ```
   outp(base+15,0x04);
   ```

   Counter 1:

   ```
   outp(base+15,0x84);
   ```

Read a Counter Sequence

1. Latch the counter.

   Counter 0:

   ```
   outp(base+15,0x40);
   ```

   Counter 1:

   ```
   outp(base+15,0xC0);
   ```

2. Read the data.

   The value is returned in 3 bytes, low, middle, and high (2 bytes for counter 1).

   Counter 0:

   ```
   low=inp(base+12);
   ```

   Counter 1:

   ```
   low=inp(base+12);
   ```

```
middle=inp(base+13);    high=inp(base+13);
high=inp(base+14);
```

3. Assemble the bytes into the complete counter value.

Counter 0:	Counter 1:
`val = high * 2^16 + middle * 2^8 + low;`	`val = high * 2^8 + low;`

Disabling the Counter Gate Command

Disabling the counter gate, as shown below, causes the counter to run continuously.

Counter 0:	Counter 1:
`outp(base+15,0x20);`	`outp(base+15,0xA0);`

Clearing a Counter Sequence

Clear a counter to restart an operation. Normally, a counter is only cleared after stopping (disabling) and reading the counter. If you clear a counter while it is enabled, it continues to count incoming pulses so the counter value may not remain at zero.

1. Stop (disable) the counter.

Counter 0:	Counter 1:
`outp(base+15,0x08);`	`outp(base+15,0x88);`

2. Read the data (optional).

   The value is returned in 3 bytes, low, middle, and high (2 bytes for counter 1).

Counter 0:	Counter 1:

`low=inp(base+12);`
`middle=inp(base+13);`
`high=inp(base+14);`
	`low=inp(base+12);`
	`high=inp(base+13);`

3. Clear the counter.

Counter 0:	Counter 1:
`outp(base+15,0x01);`	`outp(base+15,0x81);`

Watchdog Timer Programming

The Helios board has two software-programmable watchdog timers. The timers can be either BIOS-controlled using the Chipset-SouthBridge Configuration-WatchDog Configuration Menu, or software-programmable using the DSCUD 6.00 driver. Diamond Systems recommends programming the timers using the DSCUD API.

Both watchdog timers can be configured for different functionality when the timer expires. Watchdog timings are configurable from 1 second to 512 seconds in the following increments:

- 1 second
- 2 seconds
- 4 seconds
- 8 seconds
- 16 seconds
- 32 seconds
- 64 seconds
- 128 seconds
- 256 seconds
- 512 seconds

The watchdog timer can also be programmed to perform a specific action when the timer expires. The recommended action is to reset the system but the timer may also be programmed to send an NMI or generate any of the interrupts from IRQ3 to IRQ15.

Programming the Watchdog Timers Using the DSCUD API

The Helios board support in DSCUD 6.00 provides the universal interface for watchdog functionality so the application does not need to configure any of the internal registers.

> **Note: When programming the watchdog timers using DSCUD API functions, the timers should be configured as Disabled in the BIOS.**

The following watchdog timer functions are available:

Function Name	***Description***
dscWatchdogEnable	Enable both watchdog timers, specifying timeout values for each timer, independently, and the action to take when a timeout occurs. **Note: The timeout action applies to both watchdog timers.**
dscWatchdogDisable	Disable both watchdog timers.
dscWatchdogTrigger	This function **must** be called after calling *dscWatchdogEnable()*. Watchdog timers that were enabled by a previous call to *dscWatchdogEnable()* are triggered by this function. When the application calls this function, the watchdog timer is reset and a watchdog timeout does not occur. This function must then be called, periodically, by the application to prevent the watchdog from timing out.

dscWatchdogEnable Function

Function *dscWatchdogEnable()* enables both watchdog timers, specifying timeout values for each timer, independently, and the action to take when a timeout occurs. The timeout action applies to both watchdog timers.

Synopsis

```
dscWatchdogEnable ( DSCB board, WORD wd1, BYTE wd2, SDWORD options );
```

Parameters

Name	***Data Type***	***Description***
board	*DSCB*	Current board pointer obtained from previous call to *dscInitBoard()*.
wd1	*WORD*	Timeout settings for watchdog 1. This is the timeout interval before watchdog timer 1 expires, if function *dscWatchdogTrigger()* is not called before the timer expires. If the watchdog timer does expire, the action specified by the *option* parameter occurs. For example, if the recommended *option* System Reset (0x0D) is specified, the board will reset when the timer expires. The following table shows the mapping between the value supplied for parameter *wd1* and the actual time set in the watchdog timer. 0 = Disabled 1 = 1 second 2 = 2 seconds 3 = 4 seconds 4 = 8 seconds 5 = 16 seconds 6 = 32 seconds 7 = 64 seconds 8 = 128 seconds 9 = 256 seconds 10 = 512 seconds
wd2	*BYTE*	Timeout settings for watchdog 2. This is the timeout interval before watchdog timer 1 expires, if function *dscWatchdogTrigger()* is not called before the timer expires. If the watchdog timer does expire, the action specified by the *option* parameter occurs. For example, if the recommended *option* System Reset (0x0D) is specified, the board will reset when the timer expires. The following table shows the mapping between the value supplied for parameter *wd2* and the actual time set in the watchdog timer. 0 = Disabled 1 = 1 second 2 = 2 seconds 3 = 4 seconds 4 = 8 seconds 5 = 16 seconds 6 = 32 seconds 7 = 64 seconds 8 = 128 seconds 9 = 256 seconds

Name	*Data Type*	*Description*
		10 = 512 seconds
options	*SDWORD*	Configurable action upon watchdog timer expiration. This field can be any value defined in the table, below. However, it is recommended that the value of this parameter be set to System Reset (0xD0). **Note: The specified action applies to both watchdog timers.** 0x00 = Reserved 0x10 = generate IRQ3 interrupt 0x20 = generate IRQ4 interrupt 0x30 = generate IRQ5 interrupt 0x40 = generate IRQ6 interrupt 0x50 = generate IRQ7 interrupt 0x60 = generate IRQ9 interrupt 0x70 = generate IRQ10 interrupt 0x80 = N/A 0x90 = generate IRQ12 interrupt 0xA0 = generate IRQ14 interrupt 0xB0 = generate IRQ15 interrupt 0xC0 = generate NMI 0xD0 = System Reset (Recommended default action.) 0xE0-0xF0 = Reserved

dscWatchdogDisable Function

Function *dscWatchdogDisable()* disables both watchdog timers.

Synopsis

```
dscWatchdogDisable ( DSCB board );
```

Parameters

Name	*Data Type*	*Description*
board	*DSCB*	Current board pointer obtained from previous call to *dscInitBoard()*.

dscWatchdogTrigger Function

Function *dscWatchdogTrigger()* triggers the watchdog timers previously enabled by calling *dscWatchdogEnable()*.

When the application calls this function, the watchdog timer is reset and a watchdog timeout does not occur. This function must then be called, periodically, by the application to prevent the watchdog from timing out.

Synopsis

```
dscWatchdogTrigger ( DSCB board );
```

Parameters

Name	*Data Type*	*Description*
board	*DSCB*	Current board pointer obtained from previous call to *dscInitBoard()*.

Watchdog Timer Programming Example

The following is an example C program that uses the watchdog functionality on Helios board.

```
int main (void)
{
    //=========================================================================
    // I. DRIVER INITIALIZATION
    //
    //    Initializes the DSCUD library.
    //
    //=========================================================================

    if( dscInit( DSC_VERSION ) != DE_NONE )
    {
        dscGetLastError(&errorParams);
        fprintf( stderr, "dscInit error: %s %s\n",
            dscGetErrorString(errorParams.ErrCode), errorParams.errstring );
        return 0;
    }

    //=========================================================================
    // II. BOARD INITIALIZATION
    //
```

```
    //     Initialize the HELIOS board. This function passes the various
    //     hardware parameters to the driver and resets the hardware.
    //
    //=========================================================================

    printf( "\nHELIOS BOARD INITIALIZATION:\n" );

    dsccb.base_address = 0x300;

    dsccb.int_level = 5;

    if(dscInitBoard(DSC_HELIOS, &dsccb, &dscb)!= DE_NONE)
    {
        dscGetLastError(&errorParams);
        fprintf( stderr, "dscInitBoard error: %s %s\n",
            dscGetErrorString(errorParams.ErrCode), errorParams.errstring );
        return 0;
    }

    // SET WATCHDOG TIMER 1 TIMEOUT value to 64 Seconds and
    // WATCHDOG TIMER 2=DISABLED

    dscWatchdogEnable ( dscb , 7 , 0 , 0xD0 ) ; // Action = Reset board

    while ( 1 )
    {
        // other application specific actions…
        dscWatchdogTrigger ( dscb ) ;
    }

    return DE_NONE ;
}
```

Note: If the *dscWatchdogTrigger (dscb)* call is removed from the above example code, the watchdog timer will reset the system after 64 seconds.

FlashDisk Module

Helios is designed to accommodate an optional solid-state FlashDisk module. This module contains 128MB to 4GB of solid-state non-volatile memory that operates like an IDE drive without requiring additional driver software support.

Model	*Capacity*
FD-128-XT	128MB
FD-256-XT	256MB
FD-512-XT	512MB
FD-1G-XT	1GB
FD-2G-XT	2GB
FD-4G-XT	4GB

Figure 14: FlashDisk Module

Installing the FlashDisk Module

The FlashDisk module installs directly on the IDE connector, J12, and is held down with a spacer and two screws onto a mounting hole on the board.

The FlashDisk module contains a jumper for master/slave configuration. For master mode, install the jumper over pins 1 and 2. For slave mode, install the jumper over pins 2 and 3.

Configuration

To configure the CPU to work with the FlashDisk module, enter the BIOS by pressing F2 during startup. Select the Main menu, and then select *IDE Primary Master*. Enter the settings shown in the following table.

Setting	*Value*
Type	User
Cylinders	977 for 128MB flashdisk
Heads	8 for 128MB flashdisk
Sectors	32 for 128MB flashdisk
Multi Sector Transfer	Disable
LBA Mode Control	Enable
32 Bit I/O	Disable
Transfer Mode	Fast PIO 1
Ultra DMA Mode	Disable

Exit the BIOS and save the change. The system will now boot and recognize the FlashDisk module as drive *C:*.

Using the FlashDisk with Another IDE Drive

The FlashDisk occupies the board's 44-pin IDE connector and does not provide a pass-through connector. To utilize both the FlashDisk and a notebook drive, the Diamond Systems ACC-IDEEXT adapter and cables are required.

Power Supply

The 44-pin cable carries power from the CPU to the adapter board and powers the FlashDisk module and any drive using a 44-pin connector, such as a notebook hard drive.

A drive utilizing a 40-pin connector, such as a CD-ROM or full-size hard drive, requires an external power source through an additional cable. The power may be provided from the CPU's power out connector, J5, or from one of the two 4-pin headers on the ACC-IDEEXT board. Helios cable no. 6981006 may be used with either power connector to bring power to the drive.

FlashDisk Programmer Board

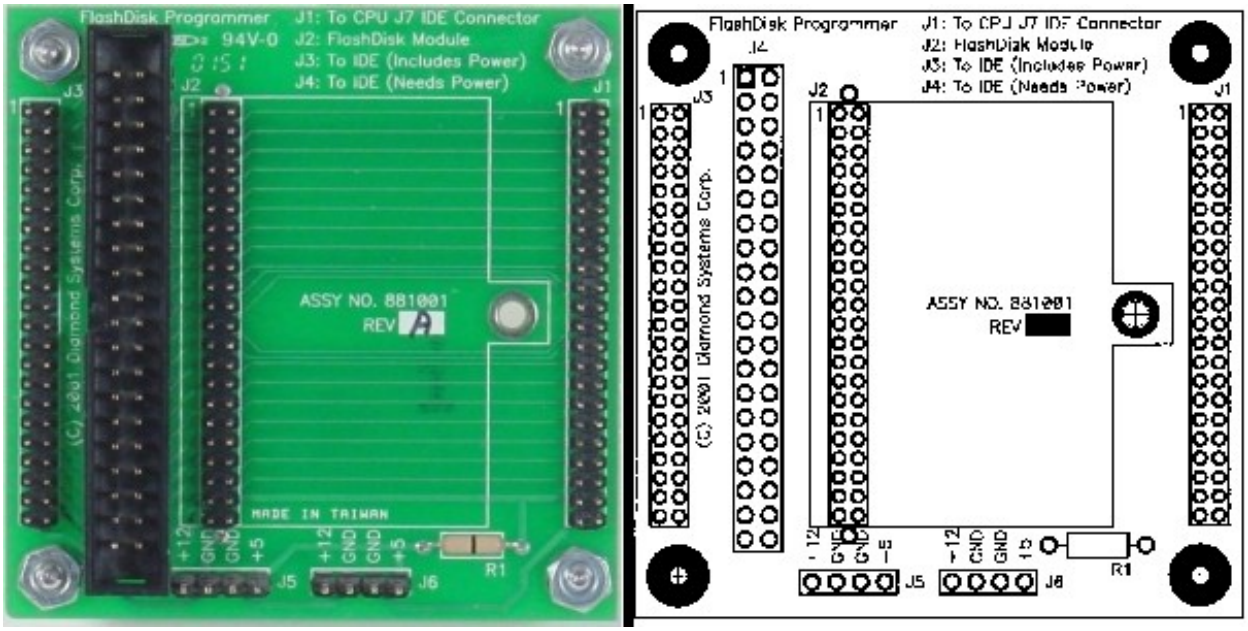
The FlashDisk Programmer Board accessory, model no. ACC-IDEEXT, may be used for several purposes. Its primary purpose is to enable the simultaneous connection of both a FlashDisk module and a standard IDE hard drive or CD-ROM drive, to allow file transfers to/from the FlashDisk. This operation is normally done at system setup. The board can also be used to enable the simultaneous connection of two drives to the SBC.

Connector J1 connects to the IDE connector on Helios with a 44-pin ribbon cable (Diamond Systems part no. 6981004). Both 40-pin .1-inch spacing, J4, and 44-pin 2mm spacing, J3, headers are provided for the external hard drive or CD-ROM drive. A dedicated connector, J2, is provided for the FlashDisk module. Any two devices may be connected simultaneously using this board with proper master/slave jumper configurations on the devices.

The FlashDisk Programmer Board comes with a 44-wire cable no. (DSC no. 6981004) and a 40-wire cable no. (DSC no. C-40-18) for connection to external drives. The FlashDisk module is sold separately.

The 44-pin connector (J1, J2 and J3) and mating cable carry power, but the 40-pin connector (J4) and mating cable do not. Connectors J5 and J6 on the accessory board may be used to provide power to a 44-pin device attached to the board when the board is attached to a PC via a 40-pin cable. These headers are compatible with the floppy drive power connector on a standard PC internal power cable.

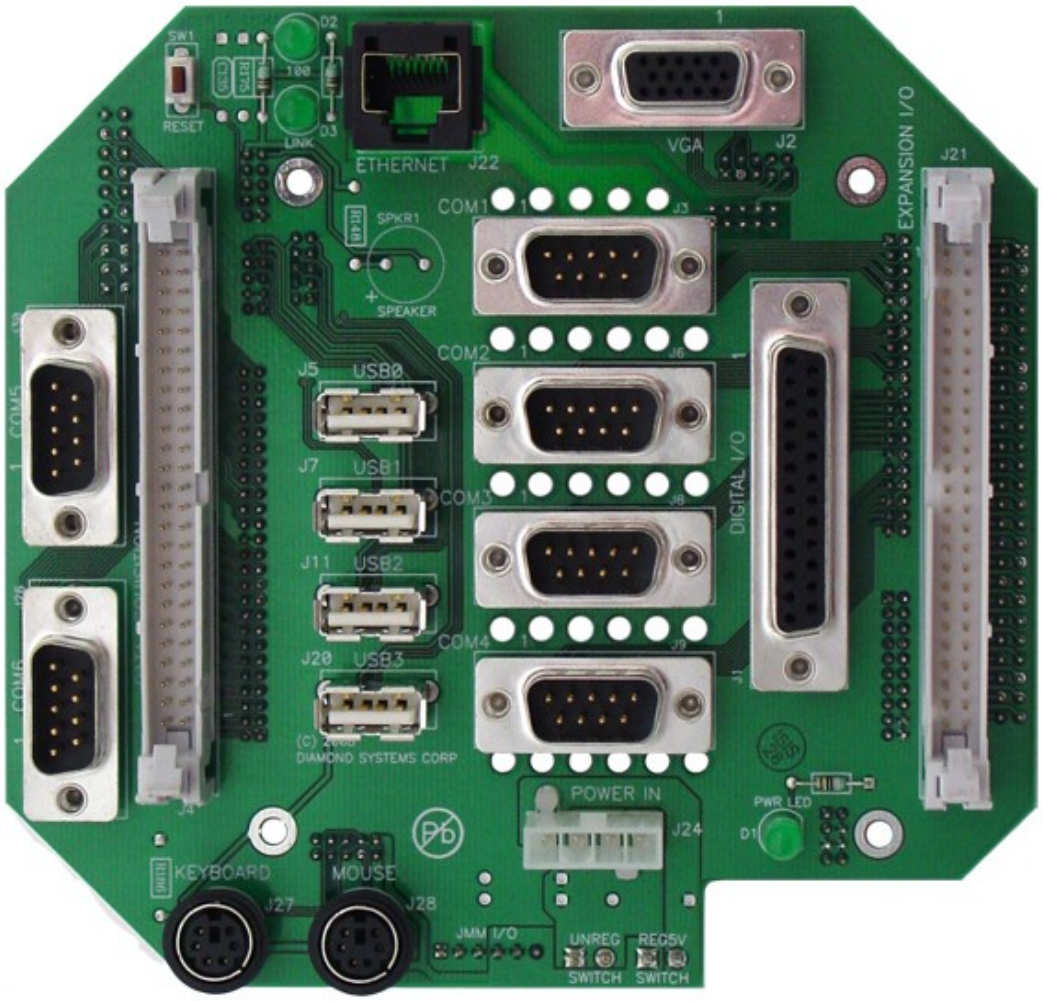
Figure 15: FlashDisk Programmer Board Layout



Panel I/O Board

Figure 16 shows the panel I/O board that mates to the main Helios board.

Figure 16: Helios Panel I/O Board



I/O Cables

Diamond Systems offers cable kit C-HLV-KIT with the following cables to connect to all I/O headers on the board. Some cables are also available separately.

Figure 17: Helios Cables Kit (C-HLV-KIT)

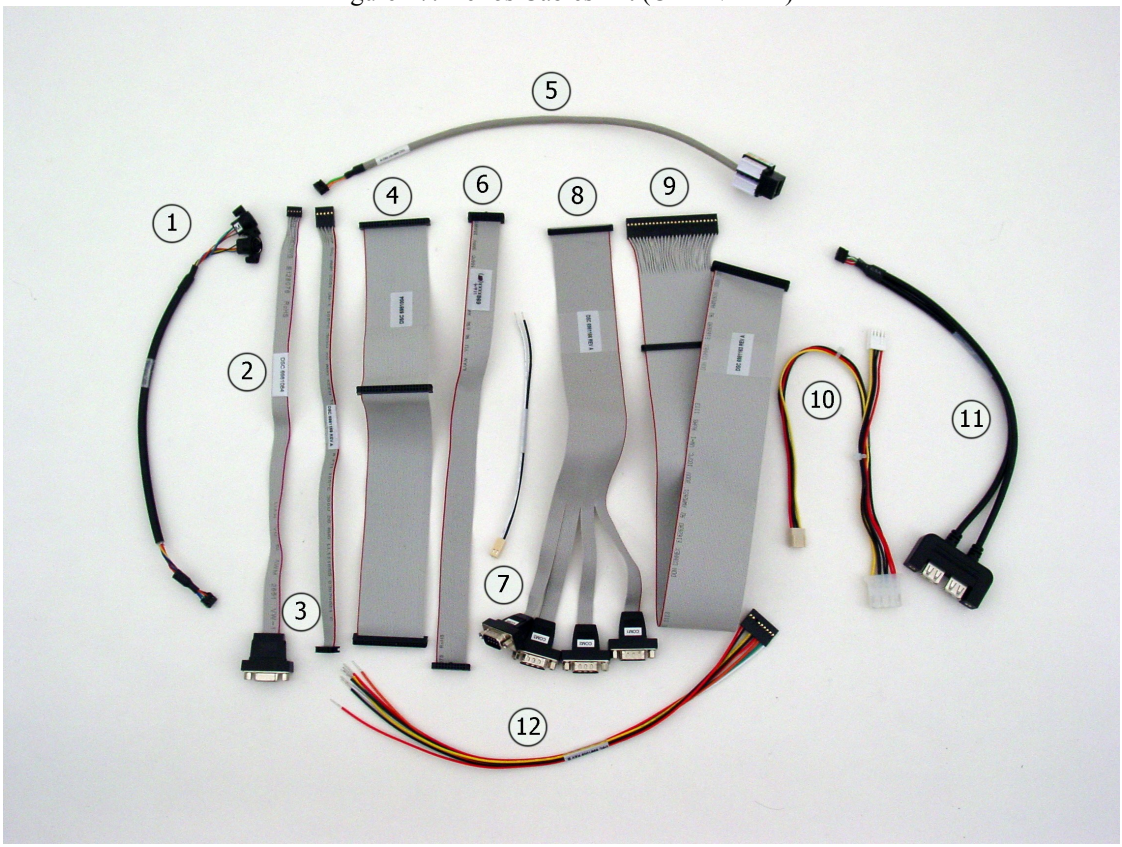


Photo No.	*Cable No.*	*Description*
1	6981083	Keyboard/Mouse (J3)
2	6981084	VGA (J10)
3	6981169	Auxiliary (J14)
4	6981004	HDD, IDE (J12)
5	6981161	Ethernet RJ45 (J11)
6	6981164	Digital I/O (J7)
7	6981180	External Battery (J6)
8	6981166	Serial Ports 1-4 (J8)
9	6981163	Data Acquisition (J17)
10	6981006	Power Out (J5)
11	6981082	Dual USB (J15, J16)
12	6981009	Power I/O (J4)

Specifications

CPU

- Processor: Vortex86DX/SX
- Speed: 300-800MHz
- Power consumption: 3.5W
- Cooling: Heat sink with fan
- Operating Temperature: -40 to +85ºC
- System Bus: 100MHz
- SDRAM memory: 128MB or 256MB 533MHz DDR2 soldered on-board
- Bus interface: PC/104 (ISA)
- Display type: CRT and/or 24-bit dual channel LVDS flat panel
- CRT resolution: 1600 x 1200
- Flat Panel Resolution : UXGA 1600 x 1200
- Video memory: 128MB UMA
- USB ports: (4) USB 2.0
- Serial ports: (2) RS-232 and (2) RS-232/422/485
- Networking: 10/100Base-T Ethernet
- Mass storage interfaces: (1) IDE UDMA 100, flashdisk interface
- Keyboard/mouse: PS/2
- Audio: AC '97, Line-in, Line-Out, Mic and amplified speaker interface

Analog Inputs

- No. of inputs: 8 differential or 16 single-ended (user selectable)
- A/D resolution: 16 bits (1/65,536 of full scale)
- Input ranges,
  - Bipolar: ±10V, ±5V, ±2.5V, ±1.25V
  - Unipolar: 0-10V, 0-5V, 0-2.5V
- Input bias current: 50nA max
- Maximum input voltage: ±10V for linear operation
- Over-voltage protection: ±35V on any analog input without damage
- Nonlinearity: ±3LSB, no missing codes
- Drift: 10PPM/^{o}C typical
- Conversion rate: 100,000 samples per second max
- Conversion trigger: software trigger, internal pacer clock, or external TTL signal
- FIFO: 512 samples, programmable interrupt threshold

Analog Outputs

- No. of outputs: 4
- D/A resolution: 12 bits (1/4096 of full scale)
- Output ranges,
  - Unipolar: 0-10V or user-programmable
  - Bipolar: ±10V or user-programmable
- Output current: ±5mA max per channel
- Settling time: 4µS max to ±1/2 LSB
- Relative accuracy: ±1 LSB
- Nonlinearity: ±1 LSB, monotonic

Digital I/O

- No. of lines: up to 40
- Compatibility: 3.3V and 5V logic compatible
- Input voltage: Logic 0: -0.5V min, 0.8V max; Logic 1: 2.0V min, 5.5V max
- Input current: ±1µA max
- Output voltage: Logic 0: 0.0V min, 0.4V max; Logic 1: 2.4V min, 3.3V max

- Output current: Logic 0: 12mA max; Logic 1: -8mA max
- I/O capacitance: 10pF max

Counter/Timers

- A/D pacer clock: 24-bit down counter
  - Clock source: 10MHz, 1MHz or external signal
- General purpose: 16-bit down counter
  - Clock source: 10MHz, 100KHz or external signal

Power Supply

- Input Voltage: +5VDC ±5%

General

- Dimensions: 3.550" x 3.775" (90mm x 96mm)
- Weight: 2.5 oz./70.8 g

Additional Information

Additional information can be found at the following websites.

- Diamond Systems Corporation: http://www.diamondsystems.com/

Technical Support

For technical support, please email support@diamondsystems.com or contact Diamond Systems technical support at 1-650-810-2500.